

CHALMERS TEKNISKA HÖGSKOLA

EENX16 KANDIDATARBETE

**Fallolyckor - Data fusion och UI/UX för en
tryggare tillvaro för anhöriga till äldre
inom hemsjukvården**

Pegah AMANZADEH
Magnus BRYNTHESON
Tilde KVARNSTRÖM
Ingrid STAKE
Maja VIKLA

10 maj 2023

Abstract

During the year of 2020, more than 100 000 people in the ages 65 or older required care due to a fall accident. The most severe fall injuries include hip fractures and intracranial injuries which generates a significant cost for society. The average cost for a person with a severe fall injury is estimated to be a one-time cost of approximately 273 600 SEK. Fall accidents can have significant consequences for the elderly in society including loss of independence, reduced quality of life and in some cases, death. Today, it can take several hours before an elderly person receives help from healthcare workers after a fall has occurred. It is crucial for elderly individuals to receive prompt care to increase their chances of survival and limit the long-term effects such an accident can have on a person's life, since nearly 50 percent of elderly individuals who have fallen cannot get up without assistance. Considering every fifth person in Sweden provide support or care to a family member, relatives play a big part in the healthcare of elderly. A major contributing factor to improve the situation is to involve relatives of the elderly to enable a quicker care process when a fall has occurred. This thesis addresses this challenge by proposing and theoretically evaluating a mobile application solution with including sensors for automatic fall detection. The proposed solution would inform relatives through a mobile application by automatically detecting a fall from the wearable sensor unit on the elderly, displaying information such as location and whether help is on the way directly in the application. Literature studies on the impact of fall accidents on society and how to design a user friendly application was conducted. Further, interviews were performed with experienced people in healthcare services to understand the current scenarios and care chain following a fall accident in the home of an elderly person. User testing was performed to gain knowledge of how well the application performs out of the user's perspective to evaluate its usability. The collected information was used design an application for the relatives of the elderly, which was then developed using *Expo*, a framework to build *Reactive Native* apps for the frontend and *Express.js*, a framework built on *Node.js* for the backend.

Sammandrag

Under året 2020 behövde över 100 000 personer i åldern 65 år eller äldre vårdas på grund av en fallolycka. De allvarligaste fallskadorna inkluderar höftfrakturer och intrakraniella skador, vilket genererar stora kostnader för samhället. I genomsnitt estimeras kostnaden för en svårt fallskadad person till en engångskostnad på cirka 273 600 SEK. Fallolyckor kan ha betydande konsekvenser för äldre i samhället såsom förlust av självständighet, nedsatt livskvalitet och i vissa fall dödsfall. Idag kan det ta flera timmar innan en äldre person får hjälp från vården efter att ett fall har skett. Det är avgörande för äldre att få snabb vård för att öka deras chanser till överlevnad, eftersom nästan 50 % av äldre personer som har fallit inte kan resa sig upp utan hjälp. Med tanke på att var femte person i Sverige ger stöd eller vård till en familjemedlem utgör anhöriga en stor roll i äldreomsorgen. En bidragande faktor för att förbättra situationen är att involvera äldres anhöriga för att möjliggöra en mer inkluderande och tryggare vårdprocess när en fallolycka har inträffat. Denna uppsats tar upp denna utmaning genom att föreslå och teoretiskt utvärdera en mobilapplikationslösning med tillhörande sensorer för automatisk falldetektering. Den föreslagna lösningen ska informera anhöriga via en mobilapplikation genom att automatiskt upptäcka ett fall från den bärbara sensorenheten på den äldre personen och visa information som plats och om hjälp är på väg direkt i applikationen. Litteraturstudier om fallolyckors påverkan på samhället och hur man utformar en användarvänlig applikation genomfördes. Vidare utfördes intervjuer med personer inom hälso- och sjukvården för att förstå den nuvarande äldreomsorgen och vårdkedjan efter en fallolycka har skett i hemmet. Användartester genomfördes för att få kunskap om hur applikationen presterar ur användarens perspektiv för att utvärdera dess användbarhet. Den insamlade informationen användes för att designa en applikation för anhöriga till äldre. Applikation utvecklades sedan genom att använda *Expo*, ett ramverk till *React Native* för frontenden och *Express.js*, ett ramverk för *Node.js* för backenden.

Ordlista

Användarcentrerad design - Ett etablerat tillvägagångssätt vid design av gränssnitt där den tänkta användaren och dennes behov och krav är i fokus.

Användbarhet - Ett begrepp som talat om hur enkelt och effektivt användare kan interagera med en produkt eller ett system för att uppnå sina mål.

Falldetektering - En teknik som används för att automatiskt upptäcka om en person har fallit, genom exempelvis sensorer.

FHIR - Ett standardiserat format för utbyte av data inom sjukvård.

Normans sju designprinciper - Sju generella riktlinjer vid design av en produkt.

Primär målgrupp - Innefattar anhöriga till personer som befinner sig i riskgrupp och bär falldetekterande sensorer.

Sekundär målgrupp - Innefattar de personer som befinner sig i riskgrupp och bär falldetekterande sensorer.

Universell design - Innebär att en design är inbjudande och användbar för en bred och varierad målgrupp.

Innehåll

1	Falolyckor och dess konsekvenser för samhället	1
1.1	Vårdkedjan	1
1.2	Falldetektering med hjälp av sensorer	2
1.3	Anhörigas roll	4
1.4	Syfte	4
1.4.1	Frågeställning	4
1.5	Mål och avgränsningar	5
1.6	Etik	5
2	Design av användargränssnitt	6
2.1	Användarcentrerad design	6
2.2	Användbarhet och universell design	7
2.3	Design för mobiltelefoner	9
2.4	Normans sju designprinciper	9
2.4.1	Använd kunskap både från verkliga livet och från minnet	10
2.4.2	Förenkla strukturen av uppgifter	10
2.4.3	Framhäv det viktiga	11
2.4.4	Strukturera rätt	11
2.4.5	Utnyttja begränsningar, både naturliga och artificiella	12
2.4.6	Designa för fel	12
2.4.7	När allt går fel, standardisera	12
2.5	Design av text	12
2.6	Generell estetik	13
2.7	Systemarkitektur	14
2.7.1	Databasalternativ och standarder inom sjukvården	14
2.7.2	Frontend	16
2.7.3	Backend	16
3	Metod	17
3.1	Datainsamling	17
3.1.1	Litteraturstudie	17
3.1.2	Intervjuer	17
3.1.3	Enkät	18
3.1.4	Användartestning av design	19
3.2	Design	20
3.3	Val av Systemarkitektur	21
3.4	Utveckling av backend	21
3.5	Utveckling av frontend	23

4	Resultat	27
4.1	Resultat av intervjuerna	27
4.2	Resultat av enkäten	27
4.3	Resultat av användartestning	30
4.4	Resultat av Design	30
5	Diskussion	35
5.1	Studiens bidrag	35
5.2	Designprocessen	36
5.3	Gränssnittsdesign	36
5.4	Användarfeedback och tester	38
5.5	Etiska aspekter	38
5.6	Tekniska utmaningar	39
5.7	Begränsningar	39
5.8	Framtida arbete	40
6	Slutsats	42
	Bilagor	45

Kapitel 1

Fallolyckor och dess konsekvenser för samhället

I Sverige dör runt 1000 personer 65 år och äldre varje år av en fallolycka och antalet dödsfall har fortsatt ökat sedan år 2000 [1], [2]. Ett fall definieras som en händelse när en person oavsiktligt ramlar ner på golvet eller marken, vilket kan leda till konsekvenser i form av fallskador [2]. Fallolyckor är händelser som kan ge upphov till stora hälsorisker, speciellt för äldre. En fallskada är en fysisk skada som uppstår i samband med ett fall och kan även ha ytterligare konsekvenser i form av känslomässigt lidande. Under året 2020 behövde mer än 100 000 svenskar i åldern 65 år eller äldre vårdas på grund av en fallskada [1]. De allvarligaste fallskadorna innefattar höftfrakturer och intrakraniella skador [3]. Nästan 50 procent av de äldre som ramlat kan inte resa sig upp utan hjälp [4]. Potentiella konsekvenser av att en äldre person blir liggandes på golvet eller marken i mer än en timme är exempelvis rabdomyolys, lunginflammation, trycksår och uttorkning. Med tanke på att medellivslängden har ökat med 20 år sedan 1960-talet och fortsätter att göra det, krävs utveckling av system inom falldetektion för att minska dödsfallen efter att en fallolycka har skett [5], [6]. Inte minst genererar fallolyckor stora kostnader för samhället. Den genomsnittliga kostnaden för en svårt fallskadad person estimeras till en engångskostnad på cirka 273 600 kronor [7]. Detta gör falldetektering ytterst aktuellt för att minska dessa dödsfall, säkerställa hälsosamma liv och främja välbefinnande för personer i alla åldrar, vilket är FN:s tredje mål i de 17 globala målen för hållbar utveckling [5].

1.1 Vårdkedjan

I dagens äldrevårdsomsorg används trygghetstelefoner som ett sätt att komma i kontakt med personal från Trygghetsjourens larmcentral vid oförutsägbara händelser i hemmet [8]. Fallolyckor är exempel på sådana händelser, även om det inte specifikt måste handla om det. Trygghetstelefonerna har en högtalarfunktion och ansluts genom ett eluttag. Dessutom får vårdtagarna en larmknapp som kan bäras på handleden eller runt halsen som trycks in när man vill larma. Då rings personalen på larmcentralen upp via trygghetstelefonens högtalarfunktion och personen kan börja berätta vad som behövs hjälp med. I Göteborg kommer det in cirka 1000 larm per dygn [9], mer än 41 larm i timmen, vilket gör det finns stor risk att flera larmar samtidigt. Detta kan resultera i en ökad väntetid vilket gör att hemtjänstens larmenhet prioriterar personer med allvarligast skador först. Därmed kan personer bli liggandes på marken efter fallet i väntan på hjälp att ta sig upp. Larmcentralen är inte ett akutlarm och tar inte rollen som larmnumret 112.

Genom att urskilja olika signifikanta händelser i trygghetstelefonernas vårdprocess kan verksamhetsförbättringar identifieras. På så sätt kan ett helhetsunderlag för utvecklingspunkter

skapas. En av dessa identifierade punkter är att vid användandet av trygghetstelefoner idag tillkommer ett flertal procedurer som behöver följas innan en person får hjälp [10]. Procedurerna illustreras i Figur 1.1 som visar de olika stegen i vårdkedjan.



Figur 1.1: Illustration av vårdkedjan.

När en person faller i sitt hem och larmar via sin larmknapp kommer larmet till Trygghetsjourens larmcentral och personen kan sedan berätta vad som inträffat [9]. Larmoperatören skickar därefter händelsen via ett meddelande i applikationen *Symphonie* till larmenheten i stadsdelen där personen bor. Därefter gör larmenheten ett bedömning om de ska kontakta hemtjänst som åker till platsen eller om de själva gör det. Väl på plats kontaktar larmenheten eller hemtjänsten som har åkt till platsen 112 eller annan vårdgivare om behovet finns.

1.2 Falldetektering med hjälp av sensorer

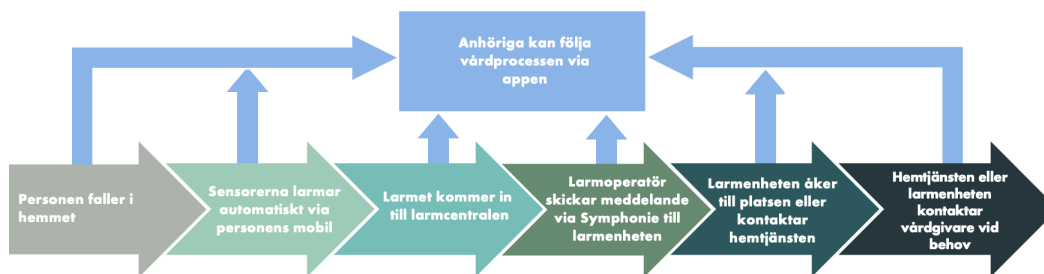
Ett sätt att förbättra förutsättningarna för äldre som fallit är att korta ner tiden från fall till vård. Detta kan uppnås genom att i realtid detektera när ett fall har inträffat med hjälp av sensorer. Sensorerna kan vara inbyggda i bärbara system såsom smartklockor och fästs på kroppen för datainsamling [11]. Apple Watch är ett exempel på ett bärbart system med en digital skärm och inbyggda sensorer som sitter runt handleden, se Figur 1.2 [12]. Modellerna Apple Watch SE, Series 4 och de senare modellerna är smartklockor som kan detektera fallolyckor när man bär klockan genom att använda en accelerometer och gyroskop [13]. När klockan detekterar ett fall vid de plötsliga förändringar i hastighet och användarens orientering, kommer klockan att avge ett lätt tryck på handleden. Samtidigt kommer ett alarm utlösas och en varning på skärmen kommer att synas enligt Figur 1.2. Om Apple Watch känner av att användaren har rört på sig efter fallet väntar den på att personen ska svara på larmet och ringer inte automatiskt ett nödsamtal. Om klockan inte detekterar några rörelser på 60 sekunder rings ett nödsamtal automatiskt. När samtalet kopplats fram spelar Apple Watch upp ett ljudmeddelande som informerar larmcentralen om att klockan har detekterat ett fall. Därefter delas den aktuella platsinformation som användaren befinner sig på.



Figur 1.2: Apple Watch [12].

Några exempel på vad som kan mätas av sensorer är mängden glukos i blodet, rörelse, hjärtfrekvens och kroppstemperatur [14]. På så sätt kan exempelvis acceleration och puls mätas för att upptäcka

en snabb ökning i dess värde vilket skulle kunna antyda att ett fall har inträffat. När detta har skett skulle förslagsvis Trygghetsjourens larmcentral automatiskt ringas upp via den drabbades mobil. Det görs då en bedömning om 112, larmenheten eller hemtjänsten borde åka till platsen [10]. Fördelen med dessa bärbara system är deras förmåga att samla in data utanför laboratoriska miljöer [11]. Via sensorerna skulle man sedan kunna registrera de uppmätta värdena från kroppen in till en mobilapplikation för anhöriga att följa. Därmed skulle anhöriga kunna få en notifikation när ett fall har inträffat hos personen och följa vårdprocessen. En alternativ vårdkedja som uppkommer i och med användandet av sensorer illustreras i Figur 1.3 nedan. Detta kan göra så att tiden från fall till att anhöriga meddelas minskar, samtidigt som vårdpersonal avlastas.



Figur 1.3: Vårdkedjan till följd av användandet av sensorer.

Idag pågår ett forskningsprojekt där man utvecklar en sensor som mäter balans och rörelse [9]. Sensorn som heter Snubblometern inriktar sig mot äldreomsorgen och syftar till att användas för att förebygga och detektera fall. Idén är att sensorn ska tejpas fast på låret för att samla in data såsom rörelser och kroppsposition, se Figur 1.4. Snubblometern monitorerar, förutom vid fall, även hur man rör sig och sover. På så sätt skulle man kunna se över datan dagarna innan olyckan och också skicka den vidare till sjukgymnaster för att få en optimerad rehabilitering. När ett fall har detekterats av sensorn skickar den ett larm till personens smartklocka som sitter runt handleden som i sin tur skickas till larmcentralen där en operatör tar emot larmet. De förmedlar sedan vidare informationen till hemtjänsten som åtgärdar händelsen. Fördelen med Snubblometern är att hjälp kan anlända snabbare när en fallolycka skett tack vare den inbyggda GPS som skickar lokalisering till hemtjänstens applikation *Symphonie*. Detta gör att man inte är bunden till sitt hem på samma sätt som man är med dagens trygghetstelefoner och får på så sätt mer frihet och lugn av att vistas utomhus. Sensorn skickar kontinuerlig information till hemtjänsten om folk skulle hjälpa personen på gatan innan hemtjänsten anländer och personen möjligtvis skulle åka ambulans. Då skulle hemtjänsten kunna se det och skicka då inte personal i onödan som kostar pengar. Problematiken med Snubblometern är främst integreringen av de olika komponenterna. Det är en leverantör av smartklockan, en annan för Snubblometern och sedan Trygghetsjouren. Med tanke på att det är helt olika företag med separata system är de svåra att integrera med varandra på grund på den låga interoperabiliteten. Det största problemet är kostnaden och viljan att integrera. Detta är vad som skett med Snubblometern då produkten har funnits i 10 år och är ännu inte ute på marknaden. Ett ytterligare problem är den mängd data som samlas upp och vad man ska göra med den. Vem är det som ska sortera och äga datan? Detta dilemma kräver resurser för att kunna lösa.



Figur 1.4: Snubblometern [9].

1.3 Anhörigas roll

Som tidigare nämnt används i nuläget trygghetstelefoner för att rapportera fallolyckor där larmcentralen kan kommunicera med patienten och avgöra om hemtjänst, larmenheten eller ambulans behöver tillkallas [10]. I fallen där hemtjänst eller larmenheten kallas in finns inga rutiner där anhöriga kopplas in eller informeras om olyckan då anhöriga enbart underrättas då ambulans kopplas in. Detta kan uppfattas som en lucka i kommunikationen mellan vårdpersonal och anhöriga vilket kan skapa stress och oro hos anhöriga, med tanke på att de utgör en stor del av vården om äldre. Enligt Socialstyrelsen så ger var femte svensk stöd eller omsorg till en närstående [15]. En del av problematiken med att åtgärda denna avsaknad av kommunikation ligger i att inte belasta vårdpersonalen ytterligare då deras fokus bör ligga på patienten och inte de anhöriga. Därav skulle det vara till stor fördel om anhöriga kunde förses med information om fallolyckor i en applikation direkt från sensorenheten utan något mänskligt ingripande från vårdpersonal. Detta skapar en mindre oro hos anhöriga samtidigt som det öppnar upp möjligheten för att ingripa om kötiden för att få vård genom hemtjänstpersonal är lång.

1.4 Syfte

Syftet med projektet är att med hjälp av sensorer och tillhörande mobilapplikation informera anhöriga, den primära målgruppen, direkt när en fallolycka skett utan att ytterligare belasta vårdpersonal för att skapa en tryggare tillvaro för personer i riskgruppen samt deras anhöriga. Sensorenheten vänder sig i huvudsak till äldre eller sjuka, den sekundära målgruppen, som vid en fallolycka av någon anledning lider ökad risk att inte klara sig själva, medan den tillhörande applikationen vänder sig till dessa personers anhöriga. En kombination av olika sensorer detekterar ett fall och sedan rapporterar detta i realtid in till en mobilapplikation. Anhöriga förses då med information om händelsen, samt information om vilka åtgärder som vidtagits. Potentiella åtgärder skulle exempelvis kunna vara att det är en ambulans på väg, eller vilken sjukhusavdelning vårdtagaren befinner sig på. Med denna information tillgänglig i en applikation slipper närstående ringa runt för att nå klarhet i läget. Resultatet av projektet förväntas på så sätt hålla anhöriga informerade om vårdprocessen utan att belasta vårdpersonal ytterligare.

1.4.1 Frågeställning

Problemet som undersöks i projektet kan formuleras som följande:

1. Hur utformas en mobilapplikation med syfte att informera den primära målgruppen om att en anhörig från den sekundära målgruppen har fallit?
2. Hur kan man med hjälp av sensorer och tillhörande mobilapplikation involvera den primära målgruppen för att uppnå en effektivare vård när en person från den sekundära målgruppen råkat ut för en fallolycka utan att ytterligare belasta vårdpersonal?

3. Hur skulle den primära och sekundära målgrupperna påverkas av en lösning i form av falldetekterande sensorer och tillhörande applikation?

1.5 Mål och avgränsningar

Målet med projektet är att ta reda på vad anhöriga och personal i olika delar av vårdkedjan efterfrågar i en applikation samt att ta fram en prototyp för applikationen. Informationen som krävs samlas in främst genom litteraturstudier, en enkät och intervjuer. Applikationen ska vara användarvänlig och fungera för operativsystemen iOS samt Android. Applikationen ska inte släppas på marknaden då dagens befintliga lagar kring integritet och informationsdelning inte tas i hänsyn samt att det inom projektet enbart produceras en prototyp i form av en applikation. Applikationen ska inte inom projektet vara kompatibel med vårdens alla redan befintliga system och rutiner då resurser för detta saknas. Sensorenheten som detekterar fall och samlar in relevant information till den anhöriga skapas inte fysiskt i projektet på grund av att de sensorer som redan finns på marknaden anses vara tillräckliga för ändamålet. Systemet antas fungera tillräckligt bra för att inte felaktigt rapportera fall och inte heller missa en fallolycka. Funktionen för att larmcentral automatiskt tillkallas vid en detekterad olycka antas redan vara implementerad. För att kunna uppnå en effektivare vård finns det dessutom ett mål att åstadkomma vårdkedjan som illustreras i Figur 1.3.

1.6 Etik

I kontext av projektet kan två viktiga etiska aspekter att ha i åtanke identifieras: Integritet och hantering av personuppgifter. Projektet inleds med ett efterforskningsarbete som till viss del består av intervjuer och enkäter. För att värna om deltagarnas integritet är enkäterna anonyma. I rapporten är de enda förekommande uppgifterna i form av respondenternas svar, så länge dessa inte bedöms kunna kopplas en enskild individ.

Även i den slutgiltiga produkten som är redo för användning är integritet och säkerhet övervägt eftersom applikationen, bland annat, behandlar hälsodata och position från den sekundära målgruppen. Ingen utomstående bör kunna få tag på dessa. Utifrån projektets utformning finns det många lagar att ta hänsyn till, exempelvis patientdatalagen som beaktar vårdgivarens behandling av personuppgifter samt socialtjänstlagen som värnar om människors sociala trygghet och dess integritet [16], [17]. Dock är en av avgränsningarna för projektet att inget fokus kommer läggas på säkerhet och integritet. En ytterligare aspekt som tas ställning till är vårdtagarens integritet gentemot de närstående. Det har avvägts och beslutats om plats och insamlad hälsodata bara ska vara tillgänglig när ett fall har detekterats och inte hela tiden för att undvika att vårdtagaren känner sig övervakad.

Kapitel 2

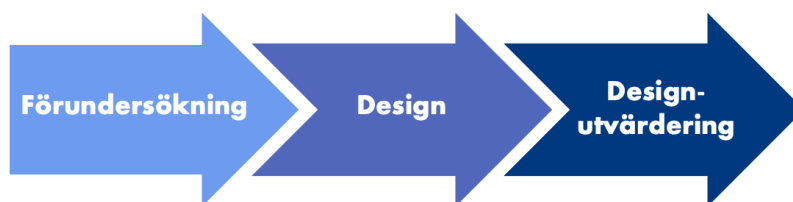
Design av användargränssnitt

Användargränssnittsdesign spelar en avgörande roll i tillverkningen av framgångsrika applikationer, webbplatser och digitala produkter. Utformningen av användargränssnittet bör inte bara vara visuellt tilltalande utan också lätt att använda och intuitivt för användarna. Ett väl utformat användargränssnitt kan öka användarnas tillfredsställelse, minska felsteg och öka produktiviteten. Därför är det viktigt för designers att följa etablerade designprinciper och mönster.

Detta avsnitt kommer ta upp Normans designprinciper, användarcentrerad design och allmänna koncept inom användargränssnittsdesign, inklusive design av text och komponenter. Det kommer också att ges exempel på hur dessa principer och koncept kan tillämpas i praktiken. Genom att förstå dessa kan designers skapa effektiva och användarvänliga gränssnitt väl anpassade till användarnas behov.

2.1 Användarcentrerad design

Det finns en rad etablerade tillvägagångssätt att tillämpa när ett gränssnitt ska tas fram [18]. Ett av dessa är *användarcentrerad design* som innefattar tre steg: Förundersökning, design och designutvärdering, se Figur 2.1.



Figur 2.1: De tre huvudstegen som ingår i en användarcentrerad designprocess.

Som namnet avslöjar kretsar alla beslut inom användarcentrerad design kring användaren [18]. Med det sagt är inte tanken att användare deltar i själva utformningen av gränssnittet, de får däremot gärna inkluderas i efterforskningarna som föregår själva designandet. På så sätt kan användarnas beteendemönster och preferenser identifieras och definieras. Med utgångspunkt från dessa observationer utformas sedan gränssnittet. Följdaktligen kan användarcentrerad design ses som ett samarbete mellan användare och designer.

Förundersökningen, som utgör den inledande fasen i en användarcentrerade designprocess ska förse designern med en förståelse för den tänkta användaren [18]. Detta görs bland annat genom att kartlägga och bilda en uppfattning om vilka användarna är och vad de har för behov. Fasen kan, i sin tur, oftast delas upp i flera steg: planering av datainsamlingen, genomförande av datainsamlingen, samt analys och rapportering av själva datan. Under planeringsfasen är målet att få klarhet i vilka relaterade affärsintressen som finns, samt identifiera vad de innebär i form av mål, begränsningar och antaganden för produkten. På så sätt kan designern fatta informerade beslut om vad som bör undersökas och vilken data som ska samlas in.

Nästa steg är att genomföra designundersökningarna [18]. Om designern inte redan är väl insatt i det aktuella området bör denne bedriva bakgrundsforskning innan några användarintervjuer inleds. Detta för att göra sig bekant med aktuella trender, termer och processer hos målgruppen. Användarintervjuerna kan genomföras på en rad olika sätt, där ett av de mest effektiva är personliga intervjuer. Även dessa kan bedrivas på flera sätt. Exempelvis genom att ställa frågor eller genom skuggande av en användare och hur de använder en applikation. Man kan även använda sig av andra metoder såsom enkätundersökningar eller insamlad data från kundtjänsten, teknisk support och marknadsundersökningar.

Beroende på vilka typer av undersökningar som bedrivits och vilken data som samlats in under föregående steg kan analysfasen ta olika former. I och med att användarcentrerad design tenderar att använda kvalitativa undersökningar går analyserna till stor del ut på att lista och bunta samman (jfr. eng. *cluster*) data för att nå resultat i form av exempelvis upptäckta trender och skillnader. När resultatet sedan rapporteras är syftet att presentera slutsatser dragna om användarna och deras behov, samt rekommendationer kring designen.

Med en god förståelse för de tänkta användarna inleds nu själva formgivandet av produkten [18]. Designfasen innefattar hjärnstormande och skissande baserat på resultatet från föregående fas, men denna process är dock dåligt dokumenterad. Därefter följer den tredje och sista fasen: Designutvärdering, som till stor del går ut på att testa användbarheten (jfr. eng. *usability*) av designen. Andra utvärderingstekniker som används är olika typer av granskningar, genomgångar och enkäter.

2.2 Användbarhet och universell design

Konceptet användbarhet är viktigt för att främja en positiv upplevelse för användaren när denne interagerar med ett mjukvarusystem [19]. Begreppet syftar på hur väl ett mjukvarusystem gör sig förstått av sina användare, samt hur den möter deras behov och förväntningar [19], [20]. Dock skiljer sig innebörden av användbarhet för en mobilapplikation och användbarhet för en webbsida. Fokus för en webbsida är att förse användaren med en rad onlinefunktioner genom ett gränssnitt som generellt lämpar sig för en datorskärm. För en mobilapplikation å andra sidan, ligger fokus snarare på att se till att användaren kan avklara mindre uppgifter (jfr. eng. *tasks*) medan denne befinner sig på rörande fot.

I [21] listas fem beståndsdelar i en användbar mobilapplikationsdesign: *Kognitivt stöd*, *informationsstöd*, *interaktionsstöd*, *prestandastöd* samt *användarstöd*. Kognitivt stöd syftar på de delar av gränssnittet som relaterar till de kognitiva aspekterna hos användaren, det vill säga hur det kan hjälpa användaren när denne tänker, kommunicerar, minns eller skapar något [19]. Informationsstöd är en benämning på de karaktärsdrag som kopplas till den information som visas för användaren [21]. Genom att exempelvis prioritera en simpel design av användargränssnittet värnar

man om användarens komfort när denne använder applikationen [19]. Detta leder i sin tur till att användaren får större självförtroende och blir mer träffsäker vid sökandet efter specifik information. Med interaktionsstöd syftar man på hur interaktionen mellan gränssnittet och användaren fungerar [21]. Här ingår exempelvis hur användaren förser systemet med input och hur systemet agerar och informerar användaren när fel uppstår [19]. Vidare avser prestandastöd mjukvarusystemets prestanda när det gäller att hjälpa till att fullfölja de uppgifter användaren tar sig an [21]. Detta är starkt kopplat till hur produkten bidrar till användarens effektivitet och tillfredsställelse i sammanhanget [19]. Slutligen riktar användarstöd in sig på hur applikationens gränssnitt hjälper de användare som exempelvis har begränsad erfarenhet av att använda en mobiltelefon. Med andra ord handlar det om hur man ser till att applikationen assisterar användare av olika skicklighetsnivåer. Detta kan till exempel röra sig om att erbjuda möjligheten att anpassa gränssnittet efter personliga preferenser för att öka produktiviteten.

Konceptet *universell design* går ut på att en produkt ska vara tillgänglig och lätt att använda för en så bred grupp som möjligt [22]. Mellan åren 1994 och 1997 genomfördes projektet "Studier för att främja utvecklingen av universell design" (*Studies to Further the Development of Universal Design*) som drevs av The Center for Universal Design på uppdrag av U.S. Department of Education's National Institute on Disability and Rehabilitation Research. En del av projektet gick ut på att ta fram riktlinjer för universell design. Detta resulterade i totalt 7 principer:

1. **Rättvis användning:** Produkten ska tilltala en bred målgrupp och erbjuda alla en likvärdig och icke-stigmatiserande upplevelse.
2. **Flexibilitet vid användning:** Designen ska erbjuda olika sätt att genomföra samma uppgift. Detta främjar anpassningsbarhet vilket i sin tur är ett sätt att göra produkter mer universellt användbara.
3. **Enkel och intuitiv användning:** Den designade produkten ska fungera på ett förväntat sätt. Den ska med andra ord vara lätt att använda och förstå oberoende av användarens erfarenhet, språkkunighet, kunskap och den aktuella koncentrationsnivån.
4. **Framträdande information:** Designen ska erbjuda olika typer av output, och på så sätt kan designen kommunicera viktig information till användaren oberoende av omgivningen och eventuella sensoriska funktionsvariationer hos användaren.
5. **Feltolerans:** Designen ska göra det svårt för användaren att begå misstag, men om det trots allt skulle hända ska felet inte resultera i skador på varken produkten eller användaren. På så sätt minimeras faror och risker för negativa konsekvenser av oavsiktliga handlingar.
6. **Låg fysisk ansträngning:** Designen kan användas effektivt och så icke-tröttsamt som möjligt. Den ska med andra ord minimera belastning och överansträngning.
7. **Storlek och utrymme för närmande och användning:** Designen är anpassad efter en variation i kroppsstorlekar och rörelseomfång. Detta innebär att lämpliga storlekar och utrymmen tillhandahålls för närmande, räckvidd, manipulation och användning oavsett användarens kroppsstorlek, hållning eller rörlighet.

Dessa sju principer har i syfte att vägleda designers att ta fram produkter som tar hänsyn till en mångfald bland användarna. De kan appliceras inom alla designdiscipliner, arkitektur såväl som mjukvarulösningar. Viktigt är dock att vara medveten om att principer endast erbjuder en utgångspunkt i designprocessen. För att kunna göra en korrekt bedömning av vilken den bäst passande designen är kräver kunskap och förståelse för det aktuella sammanhanget, samt för vilka avvägningar som behöver göras gällande tillgänglighet och användbarhet.

2.3 Design för mobiltelefoner

Mobiltelefonens begränsade skärmyta innebär utmaningar under designprocessen av tillhörande applikationer beträffande skrollande, prioriterade funktioner, färgval, text- och knappstorlek [23], [24]. Med en begränsad yta att kommunicera med användaren genom blir det viktigt att se till att ytan utnyttjas på bästa möjliga sätt. Till skillnad från horisontellt skrollande anses inte vertikalt skrollande vara särskilt betungande för användaren. Det gör att det bör ses till att elementen i så stor mån som möjligt listas vertikalt. En begränsad yta gör det också viktigt att se över vilka de absolut viktigaste funktionerna är för användaren. Om de vanligaste flödena kan identifieras, kan gränssnittet optimeras för att dessa flöden ska ta så få knapptryck och vara så enkla att utföra som möjligt. Detta innebär ofta att exempelvis mindre central funktionalitet flyttas djupare in i applikationen.

Om gränssnittet utformas för en bestämd upplösning och storlek av skärm kommer det finnas en risk för att bilder och komponenter förvrängs eller att användaren blir tvungen att skrolla horisontellt [20]. Det är alltså inte enbart att mobilskärmarna är små som är utmanande, utan även att det finns en variation av olika storlekar. Genom att implementera en responsiv design undviker man detta genom att gränssnittet automatiskt anpassar sig till den aktuella skärmens storlek och upplösning. På så sätt förbättrar man innehållets åtkomlighet och läsbarhet.

Användargränssnittsdesign är högst relevant eftersom den möjliggör säkerställandet av en smidig kommunikation mellan användaren och systemet [24]. Ett väl utformat gränssnitt kan förbättra flödet och främja aktivt engagemang hos användaren. Detta blir särskilt viktigt inom mobila enheter och tillhörande gränssnitt eftersom den begränsade skärmstorleken gör att översiktlighet är avgörande. För att uppnå detta behöver information och funktionalitet representeras av relevanta nyckelord eller andra gränssnittselement, såsom ikoner. För att minimera kognitiv belastning hos användare bör ikoner och nyckelord väljas med omsorg för att göra dem så intuitiva som möjligt.

Att kunna navigera sig runt applikationen är en av de viktigaste funktionerna. *Bottennavigering* är ett användargränssnittsmönster för att ge användarna enkel, effektiv och snabb åtkomst till de viktigaste funktionerna i applikationen [23]. Detta gränssnittsmönster visar en navigeringsmeny längs ned på skärmen, där varje menyval representeras av en ikon och textbeskrivning. Genom att trycka på en ikon kan användaren direkt nå den motsvarande vyn i applikationen. Detta användargränssnittsmönster används oftast för mobilapplikationer med ett fåtal funktioner eftersom det tillåter användaren att navigera snabbt och enkelt mellan dessa funktioner med minimal ansträngning. En annan fördel med *Bottennavigering* är att användaren kan snabbt och enkelt orientera sig och förstå vilken sektion av appen de befinner sig i genom att den aktuella ikonen är urskiljd från de andra genom att exempelvis färgas blå. Detta minskar risken för att användaren ska känna sig förvirrad i applikationen, vilket kan öka användarvänligheten. Det är därför *Bottennavigering* används i många andra applikationer såsom Instagram.

Vanemässighet (jfr. eng. *habituation*) är ett psykologiskt fenomen där användare vänjer sig vid en given stimulus med tiden [25]. Syftet med att använda vanemässighet som en typ av designmönster i applikationen är att förbättra effektiviteten och användarupplevelsen genom att förenkla utförandet av vanliga uppgifter, utan att kräva en överdriven tankeverksamhet angående processen.

2.4 Normans sju designprinciper

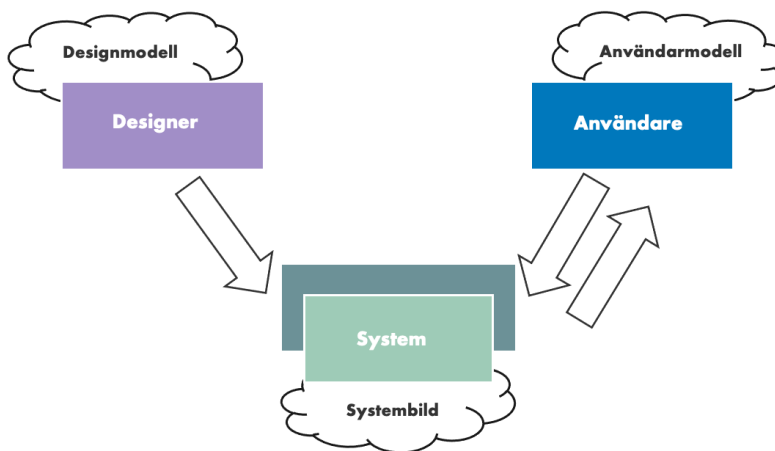
När det kommer till användarcentrerad design är det viktigt att göra det enkelt för användare att förstå vad de ska göra och hänga med på vad som händer. Detta gäller generellt, inte bara vid design av användargränssnitt [26], [27]. För att göra svåra uppgifter till enklare har Don Norman tagit fram

följande sju grundläggande designprinciper.

2.4.1 Använd kunskap både från verkliga livet och från minnet

Enligt Norman känner sig människor mer bekväma när kunskapen som krävs för att genomföra en uppgift finns tillgänglig antingen som självklarhet från verkliga livet eller läsbart härlett från tydliga begränsningar av designen [26]. Kunskap från verkliga livet är dock bara användbart om designen har en för användaren välkänd funktionalitet som kan liknas vid den. Designen ska göra det lätt för användaren att med kunskap både från verkliga livet och från minnet, utan att de stör varandra. Det som känns lättast för tillfället ska användas och designen ska tillåta en kombination av dem båda.

Det finns tre konceptuella modeller som bör tas i beaktning vid produktdesign, vare sig det rör sig om en korköppnare eller ett datasystem [26]. Dessa tre mentala modeller är designmodell, användarmodell och systembild. Designmodellen ska illustrera det som designern tänker med produkten. Användarmodellen är den modell användaren utvecklar för att förklara och förstå driften av systemet, i en perfekt värld är dessa två modeller vara ekvivalenta. Systemet självt används för kommunikation mellan designern och användaren genom dess utseende, drift, respons och instruktionerna som kommer med produkten. När det gäller systembilden behöver designern se till att den är konsekvent och skildrar driften av den rätta konceptuella modellen väl, här inkluderas även instruktioner och manualer. I Figur 2.2 illustreras de tre mentala konceptuella modellerna.



Figur 2.2: Illustration av de tre konceptuella modellerna och hur de kommunicerar med varandra.

2.4.2 Förenkla strukturen av uppgifter

Genom att minimera mängden planering och problemlösning som krävs för att lösa en uppgift förenklas strukturen [26]. Ofta kan komplexa uppgifter struktureras om med hjälp av tekniska innovationer. Som designer behöver psykologin hos användaren tas med i beräkningarna då den begränsar hur många aktiva tankar en person kan ha samtidigt och hur länge något kan hållas i minnet. En tumregel för korttidsminnet hos en person är att denne inte bör förväntas att minnas mer än fem orelaterade saker på samma gång. Detta betyder att om systemet kräver det bör systemet tillhandahålla någon slags assistans för att hjälpa användaren att minnas. När det kommer till långtidsminne är informationen lättare att uppfatta om den kan integreras till något slags konceptuellt ramverk från verkliga livet. På

så sätt kan användaren påminnas om vad som kan göras och hur det går till. Enligt Norman finns det fyra huvudsakliga tillvägagångssätt för att förenkla strukturen av uppgifter, vilka är följande:

- Behåll uppgifter så lika varandra som möjligt och tillhandahåll mentala hjälpmedel.
- Använd teknologi för att synliggöra saker som annars skulle vara osynliga för att förbättra respons och känslan av att användaren har kontroll.
- Automatisera men håll uppgifterna så enhetliga som möjligt.
- Ändra uppgiftens karaktär.

När det kommer till att behålla uppgifter så enhetliga och lika varandra som möjligt men tillhandahålla mentala hjälpmedel menar Norman exempelvis funktioner som personliga noteringar, adressbok och miniräknare i mobilen [26]. Att använda teknologi för att synliggöra saker som annars skulle vara osynliga för att förbättra respons och känslan av kontroll hos användaren kan det exempelvis röra sig om instrumentbrädan i en bil där man kan se olika delar av fordonet utan att fysiskt få tillgång till dem och utan att uppgiften för användaren förändras. Att automatisera kan vara till stor hjälp när man vill förenkla en uppgift. Dock medför det också att användaren förlorar viss kontroll. Ett exempel är användandet av automatväxlade bilar istället för manuellt växlande, då vissa föredrar att växla manuellt för att behålla kontrollen. I vissa fall är det bättre att ändra uppgiftens karaktär helt, som när en uppgift kräver vissa färdigheter för att klara av. Ett exempel på detta är skosnören då det krävs motoriska färdigheter som kanske inte alla besitter för att använda dem, då har skor med kardborreknäppning utvecklats för att förenkla uppgiften.

2.4.3 Framhäv det viktiga

Att framhäva eller synliggöra saker handlar om att göra det tydligt för användaren vilken funktionalitet produkten erbjuder, samt hur den används [26]. Användaren ska tydligt kunna se effekterna av sina handlingar. Vidare bör systemets funktioner spegla dess intentioner. Dessutom bör indikationer på systemets status vara synliga och lätta att tolka.

2.4.4 Strukturera rätt

Norman beskriver att intuitiv struktur bör eftersträvas. Det ska vara lätt för användaren att uppfatta följande relationer:

- Mellan intentioner och möjliga handlingar.
- Mellan handlingar och dess effekter på systemet.
- Mellan systemets nuvarande status och vad som kan uppfattas genom antingen syn, ljud eller känsel.
- Mellan systemets uppfattade status och behov, intentionerna och förväntningarna av användaren.

Med intuitiv struktur menar man att de spatiala relationerna mellan positionering av komponenter och systemets kontroller bör vara logisk gentemot hur systemet faktiskt fungerar [26]. Kontrollerna bör ha ett analogt förhållande eller vara placerade direkt på objekten som de är kopplade till. Samma sak gäller relationen mellan ett systems faktiska utfall och det, av användaren, förväntade utfallet. För att användaren ska kunna utvärdera om effekterna av en handling i ett system blir som önskat behövs frekvent feedback av resultaten. Information bör vara utformad på ett sådant sätt att den är lätt att förstå, samt att den stämmer överens med användarens intentioner.

2.4.5 Utnyttja begränsningar, både naturliga och artificiella

Begränsningar bör utnyttjas på ett sätt så att användaren känner att det bara finns ett enda rätt tillvägagångssätt för användandet av en produkt eller ett system [26]. Ett exempel på när begränsningar används för att reducera antalet alternativa handlingar vid varje delsteg är vid ihopbyggandet av vissa avancerade LEGO-byggnationer.

2.4.6 Designa för fel

En designer bör utgå från att alla fel som är möjliga kommer att uppstå någon gång [26]. Varje handling användaren gör kan betraktas som ett försök till att ta ett steg i rätt riktning. Ett fel kan då ses som en ofullständigt specificerad handling. Istället för att försöka utrota sådana fel, kan designern istället inkludera dem i designen genom att göra det enkelt för användaren att backa när de uppstår. Det ska vara svårt att göra irreversibla handlingar och lätt att ångra steg i oönskad riktning.

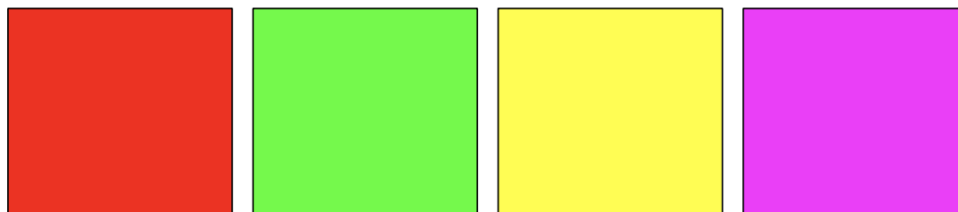
2.4.7 När allt går fel, standardisera

Att standardisera handlingar, utfall, layout och vad som visas kan vara till hjälp om något inte kan designas utan att strukturen blir slumpmässig och ologisk [26]. Genom att skapa en internationell standard, oavsett hur slumpmässig och rörig den standardiserade mekanismen blir, behöver den bara läras in en gång. Detta har till exempel applicerats på tangentbord, trafikljus, kalendrar och måtenheter. Ett annat exempel där standardisering har gjort stor nytta är när det kommer till bilar och att köra bil. Det finns standarder för vilken sida av vägen som är rätt att köra på, vilken sida förarsätet ska vara på och placering av pedaler och ratt etc. Tajming är en viktig aspekt av standardisering. Det kan vara svårt att nå överenskommelser men det är av stort värde att standardisera så tidigt så möjligt för att bespara alla parter så många problem som möjligt.

2.5 Design av text

Beroende på vilket intryck samt vilka känslor som önskas väckas hos användaren finns en rad olika aspekter att ha i åtanke när det kommer till textdesign, såsom färgval, kontrast mellan bakgrund och text, typsnitt, font och textstorlek [23], [28]. Mörkare bakgrunder brukar ge ett vassare intryck och kan uppfattas antingen mer dystert eller energiskt beroende på andra designaspekter, medan ljusare bakgrunder kan förmedla en helt annan känsla och är mer typisk för gränssnitt för datorer. Om avsikten är att förmedla ett budskap på ett lugnande och avslappnande sätt, bör kontrasten mellan texten och bakgrunden vara låg. Om avsikten däremot är att förmedla ett budskap med mer spänning och styrka kan en starkare kontrast fungera bättre. Färger med hög saturation har mycket karaktär och ger ett intryck av livlighet men kan vara uttröttade för ögonen om de överanvänds. I Figur 2.3 illustreras exempel på högsaturerade färger. Därför är det sällsynt med fler än en eller två högsaturerade färger i en och samma design. Oftast utgörs huvuddelen av mer nedtonade färger. Även olika kombinationer av färger ger olika effekter på helhetsintrycket, till exempel kan två högsaturerade färger inviga mycket högre energi och rörelse jämfört med dem var för sig. Om man avser att rikta uppmärksamheten till ett specifikt objekt på en skärm kan det med fördel ha en högsaturerad färg och omges av mer nedtonade färger. Detta kommer även ge intrycket av att objektet kommer närmare användaren. För att ytterligare förstärka ett budskap eller indikera att information är viktig bör man inte bara förlita sig på att färgvalet ska förmedla detta. En kombination av färg och form är betydande för att påvisa viktig information.

Val av typsnitt och font är av stort värde när det kommer till hur ett budskap önskas framställas [23]. Serif är en grupp typsnitt där bokstäverna har små kurvor och linjer i slutet, däribland ingår Times New Roman”. Dessa typsnitt underlättar läsningen för ögat och används ofta i sammanhang med tät



Figur 2.3: Exempel på högsaturerade färger.

text. I användargränssnitt för mobilskrmar är det dock vanligare med typsnitt utan Seriffer i slutet av bokstäverna då de gör sig bättre i mindre storlek. Ett samlingsnamn för dessa är Sans Serif. För att skapa en känsla för till exempel ett varumärke kan man använda typsnitt av typen Display, vilka fungerar väl i större storlek men lämpar sig inte för mindre format då de kan verka överväldigande och läsbarheten blir låg. Exempel på typsnitt i de ovan nämnda grupperna visas i Figur 2.4.

Serif **Sans Serif** **DISPLAY**

Figur 2.4: Exempel på typsnitt i grupperna Serif, Sans Serif och Display.

Textstorlek mäts i en sifferskala, ju lägre siffra ju mindre storlek [23]. För att ge läsaren en behaglig läsoplevelse bör textstorleken inte vara mindre än 12 i ett användargränssnitt. En funktion där möjligheten finns att justera storleken på texten till läsarens önskade storlek kan vara fördelaktigt. Radavståndet är en annan viktig aspekt när det kommer till läsbarhet. Avståndet behöver vara tillräckligt stort för att texten ska bli lätt för ögat att följa utan att skapa en känsla av att raderna är för långt separerade.

2.6 Generell estetik

På samma sätt som designen av texten i ett gränssnitt har stor betydelse för hur ett budskap uppfattas och vart användarens uppmärksamhet riktas spelar andra visuella effekter som vinklar och kurvor in [23]. Det finns vissa riktlinjer för att uppnå en bra design och i kommande avsnitt belyses hur enhetlighet kan uppnås och vilka effekter flat design, skeumorfism och minimalism kan ha i en applikation. En skärm med vertikala och horisontella linjer ger generellt ett lugnare intryck än en skärm med diagonala och oregelbundna former och vinklar. Beroende på hur man använder kurvor och former kan man skapa lugn men också addera rörelse och livfullhet. Till exempel kan ett fåtal utvalda kurvor i en annars rektangulär design rikta användarens uppmärksamhet i önskad riktning samt bidra till ett mer sofistikerat intryck. Om man vill skapa mer spänning och rörelse kan man generellt låta skärningspunkter mellan kurvor och linjer mötas i skarpa vinklar.

Enhetlighet är viktigt för att skapa en bra design då man vill skapa en känsla av att hela gränssnittet hänger ihop [23]. Att uppnå detta kan vara en utmaning men det finns några generella riktlinjer som kan vara till hjälp. En viktig faktor kan vara att upprepa visuella element och motiv,

till exempel låta kurvor och former mötas i samma vinklar genom hela gränssnittet. En annan faktor kan vara typsnittet. Att använda samma textdesign för all brödtext och samma för rubriker bidrar till enhetlighet. En fara med detta kan dock vara att användaren antar att likheterna i utseende medför likhet i funktion.

För att få en användare att förstå en funktion i ett gränssnitt kan man använda sig av skeuomorfism [23]. Detta innebär att designen härmar karaktäristiska drag från objekt i verkligheten som användaren redan är bekant med. Ett exempel är *Apple Wallet* vars ikon liknar en plånbok med kort och biljetter, vilket hjälper användaren att hitta vad de söker. Ett annat exempel är Apples kalkylatorapplikation som har utseendet av en typisk fysisk kalkylator.

Den vanligaste typen av design för webbsidor och mobilapplikationer kallas flat design [23], [29]. Designtypen karaktäriseras av enkla ikoner och logotyper och typsnitt av typen Sans Serif mot en solid bakgrundsfärg. Stilen används bland annat i transportsektorn då den skapar en universell känsla och kan inte kopplas till någon speciell kultur. Den fungerar också väl på både små och stora skärmar. Till skillnad från skeuomorfism försöker inte designen efterlikna något från verkligheten, om man bortser från ikonerna, utan snarare att smälta in i bakgrunden för att flytta användarens fokus till innehållet.

Minimalistisk design är vanlig att använda i sammanhang där data i första hand ska visas snarare än att föras in eller manipuleras av användaren [23], [30]. I dessa designer reduceras skärmelementen till minsta möjliga för att göra gränssnittet så tydligt som möjligt.

2.7 Systemarkitektur

Innan applikationen börjar byggas är det viktigt att skapa en arkitektur för databasen, frontenden och backenden. Val av databasen beror huvudsakligen på vilken data som ska sparas och vilken relation datan har till varandra. Val av språk och ramverk i frontenden väljs beroende på projektets behov samt programmerarnas tidigare kunskap. För backenden måste ett skalbart ramverk väljas för att säkerhetsställa framtida utveckling av systemet. För att få en snabb utveckling kommer även ramverk som har mycket dokumentation och bibliotek för att snabbt komma igång och utveckla applikationen. Ett problem är att sjukvården ligger bakom andra brancher i utvecklandet av IT-system och dela data mellan systemen. Största anledningen till detta är att hälsoregisterna inte är publika vilket hindrar utvecklandet och delandet av datan. Men det finns en standard som blivit mer vanlig och det är *Fast Healthcare Interoperable Resources* för hur data ska formateras och skickas mellan applikationer inom sjukvården. FHIR har skapat över 1 miljon syntetiska patient loggar för att hjälpa till i utvecklandet av nya IT system inom sjukvården utan att sprida känslig information [31].

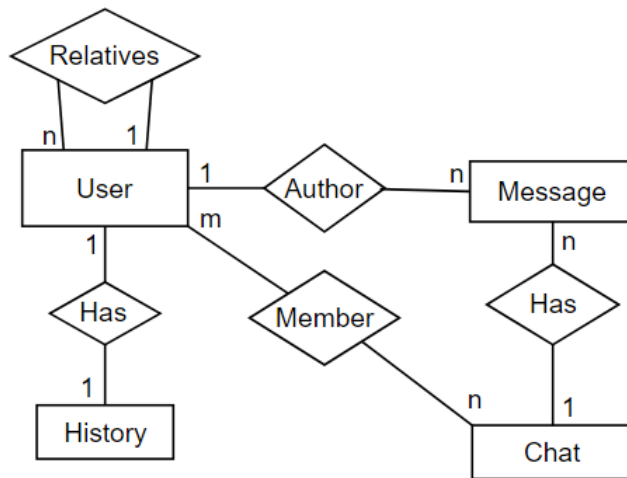
2.7.1 Databasalternativ och standarder inom sjukvården

Den internationella standarden ”FHIR”, påverkar valet av databas, enligt [32] måste databasen kunna lagra JSON-data. Det finns många relationsdatabaser idag så som *PgAdmin* som kan lagra JSON-data. Denna typ av databas är fördelaktig när datan som ska sparas har en strukturerad form.

userId	name	password	admin
1	'Test 1'	'Test'	false
2	'Test 2'	'Test'	true
3	'Test 3'	'Test'	false
4	'Test 4'	'Test'	false
5	'Test 5'	'Test'	true

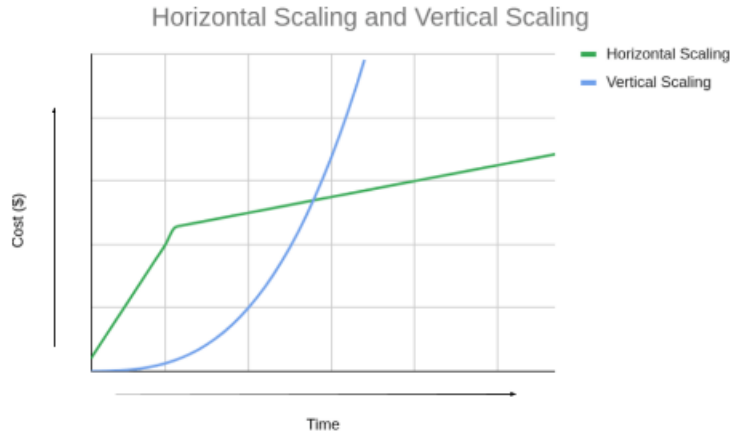
Ovan är ett exempel på hur en relationsdatabas ser ut här sparas all data i tabeller. Varje kolumn: **userId**, **name**, **password** och **admin** har en egen datatyp. I detta fall är det int, varchar, varchar och boolean. Om datan inmatas på felaktigt sätt, till exempel 'Hej' på userId kommer ett felet att det inte gick och lägga till den nya användaren att kastas.

Ett annat alternativ är *NoSql* icke-relationsdatabas (jfr. eng. *non-relation database*). Med denna typ av databas är det mycket enkelt att lagra JSON-data. *MongoDB* är den största *NoSql*-databashanteraren och erbjuder en molnbaserad version (jfr. eng. *cloud based*), detta gör att varje utvecklare inte behöver köra en lokal databas på sin dator. Detta hade varit nödvändigt om projektet genomfördes med exempelvis *PgAdmin*.



Figur 2.5: Ett exempel på hur ER-diagram över databasstrukturen skulle kunna att se ut.

I Figur 2.5 visas exempel på hur databasen kan designas. En *User* kan ha flera *Relatives*, samt en *History*, vilket är historiken över alla tidigare fall. Vidare har applikationen även en chattfunktion. Chatten kommer bestå av *Messages* och *Users*. Den här strukturen kan implementeras i antingen en relationsdatabas eller icke-relationsdatabas.



Figur 2.6: Demonstrerad kostnad för att skala horisontellt jämfört med vertikalt [33].

En fördel med en *non-sql*-databas är att den kan skalas horisontellt. Detta betyder att fler servrar läggs till och delar på belastningen av all data allt eftersom den växer. En relationsdatabas skalar vertikalt istället, vilket innebär att när mer kraft behövs, görs detta genom att lägga till resurser som *CPU* och *GPU* till servern vilket blir betydligt mer kostsamt över tid. Ytterligare en nackdel med att skala databasen vertikalt är att all data samlas på en server, medan den för en *non-sql*-databas är utspridd över flera. Detta gör att det blir lättare att återställa data ifall en server skulle gå sönder [33].

2.7.2 Frontend

Användargränssnittet är den delen av applikationen som användaren interagerar med - den del som syns av, i detta fallet, mobilapplikationen. För att programmera användargränssnittet kommer ett 'native' språk användas så att applikationen ser likadan ut oavsett operativsystem. Nativ kod syftar på maskinkod som programmet själv genererar utan att behöva översättas från någon högre nivå av källspråk. Ett alternativ är *Expo* som är ett ramverk för *React Native*. *Expo* används för att inte behöva skriva någon 'native kod' utan allt det sköts av *Expo*. *Expo* har även många egna bibliotek så som *Expo-Notifications* där man enkelt kan integrera pushmeddelanden till användarna. Detta är essentiellt när ett fall inträffar. Ett annat alternativ är *Flutter* som är Googles motsvarighet till *React Native* och använder sig av *Dart* istället för *Javascript* som programmeringsspråk. Ett CSS-ramverk kommer användas för att det ska gå snabbare att formge (jfr. eng. *style*) de olika komponenterna. *NativeWind* är ett exempel på detta, men finns många motsvarigheter så som *Bootstrap* och *Material UI*.

2.7.3 Backend

Backend är del av koden som ansvarar för att bearbeta data och kommunicera med andra system, till exempel sensorer eller larmcentralens system. En av de vanligaste typerna av Javascript-servrar är *Express.js* som är ett etablerat och dokumenterat ramverk för *Node.js*. En fördel är att *Express.js* även går att använda med *Typescript*. Detta ger en säkrare kodning då objekt blir statiskt typade istället för dynamiskt och ger kompileringssfel istället för exekveringsfel vilket är lättare för programmeraren att upptäcka. Genom detta minskar antalet kompileringssfel. *Express* har även många bibliotek såsom *Websocket* för att lätt skicka realtids data vilket är viktigt för chatt funktionen i applikationen. För att få upprätthålla ett typat projekt (frontend och backend) behövs ett *ORM*, objekt-relationell mappning (jfr. eng. *Object-Relational Mapping*). Det underlättar skrivandet av 'typad' kod som sedan kan köras på databasen. Ett exempel på *ORM* är *Prisma*, som lämpar sig särskilt väl för att koppla typer från en NoSQL-databas.

Kapitel 3

Metod

För att kunna besvara frågeställningen användes flera olika metoder för att komma fram till en applikation vid namn *SeniorSecure*. Datainsamling var en stor del av metoden för att samla in den information som krävdes för att utveckla en användarvänlig och ändamålsenlig applikation. En annan viktig del av metoden är utformning och val av design och dess användargränssnitt.

3.1 Datainsamling

Insamlingen av information kan delas in i två kategorier: primär och sekundär datainsamling. Den primära datainsamlingen, det vill säga förstahandsdata, bestod av intervjuer, användartester och en enkät till allmänheten. Den sekundära datainsamlingen är information som är insamlad av andra parter och bestod av litteraturstudier.

3.1.1 Litteraturstudie

Litteraturstudier om fallolyckors påverkan på samhället samt de redan befintliga lösningar om falldetektering utfördes för att fastställa vetenskaplig fakta och erhålla information om tekniker som används för att detektera fall. Digitala och fysiska bibliotek samt databaser såsom PubMed, Scopus, NCBI och Google Scholar användes för att förstå ämnet. Nyckelord som *fall*, *detection*, *sensors*, *accidents*, *injury*, *elderly*, *mobile interface*, *design*, *typeface*, *typography*, *UX*, *UI*, *universal design*, *usability*, *user centered design*, *design principles*, *minimalistic design*, *flat design* användes, antingen för sig själv eller kombinerat med ett eller flera andra nyckelord för att framhäva relevanta resultat från de digitala bibliotek, databaser och Googles sökmotor som användes.

3.1.2 Intervjuer

För att få en bred vy av dagens äldrevårdsomsorg samt hur man skulle kunna uppnå en effektivare vård inom fallolyckor gjordes flertalet olika intervjuer. En bidragande faktor till behov och val av intervjupersoner är de identifierade brister i litteraturstudien som behövde fyllas. I en intervju frågades en före detta vårdare inom hemtjänsten. Därför ställdes relevanta frågor angående erfarenheter inom området som inte identifierades i litteraturstudien, se nedan.

- Hur kommunicerar vårdpersonal med anhöriga efter en fallolycka i dagsläget?
- Hur hade synliggörandet av sensorvärden (puls, blodtryck osv.) i appen påverkat de anhöriga? Bör bra/dåliga värden indikeras i så fall?

- Vad hade möjligheten att chatta med de anhöriga genom appen inneburit? Hade det underlättat ert arbete?
- Vilka är de vanligaste frågorna från anhöriga ni får i samband med en fallolycka? Hur hade dessa kunnat besvaras i appen?
- Generella tankar och åsikter om appen?

Ulrica Björner, kvalitetsutvecklare för rehabilitering på Göteborgs Stad, och Martin Arvidsson, system- och driftansvarig för larmdelen för mottagaren på trygghetsjouren på Göteborgs Stad intervjuades gemensamt för att få ett perspektiv inom deras arbete. De specifika frågorna som ställdes till de båda var:

- Vad har ni för spontana tankar och idéer om vilka funktioner som borde finnas i appen?
- Hur lyder processen efter att en äldre ramlat i dagsläget? Hur tillkallas hjälp?
- Finns det någon specifik person ni tycker vi borde prata med om detta utöver målgruppen och vårdpersonal?
- Hur får vi denna app att bli så ”lockande” som möjligt för de anhöriga tror du?
- Vad tror ni är viktigast för en närstående att se vid en fallolycka (vilka egenskaper bör appen ha utöver att den ska vara lätt att använda)?

En ytterligare intervju gjordes med en pensionerad sjuksköterska inom hemsjukvården. Under denna intervju ställdes främst frågor om patienterna och den process som äger rum vid ett fall eftersom personen är erfaren inom detta område. Frågorna som ställdes var:

- Hur anpassas vården vid ett fall utifrån patienten?
- Hur lyder processen efter att en äldre ramlat i dagsläget? Hur tillkallas hjälp och så vidare?

Svaren på dessa frågor kompletterade litteraturstudien för att fånga så många perspektiv som möjligt.

3.1.3 Enkät

En enkät till allmänheten skickades ut i syfte att undersöka behovet och inställningen till applikationen och hur samhällsproblemet upplevs idag. Dessutom var syftet också att få återkoppling från den tänkta målgruppen på redan befintliga idéer om applikationens funktionalitet och vilken information som önskas. Enkäten skickades ut via både privat kontakt till personer och via en grupp på Facebook. Frågorna som besvarades i enkäten var:

1. Hur gammal är du?
2. Vad har du för kön?
3. Har du någon närstående som råkat ut för en fallolycka och behövt vård efter?
4. Om du svarade ja på föregående fråga: Hur väl upplevde du blev informerad om förloppet som följde olyckan? Hade du velat att det såg annorlunda ut? Hur hade det kunnat förbättras?
5. Har du för närvarande en anhörig som löper ökad risk för att råka ut för en fallolycka där följden kan vara långvariga skador som kräver sjukvård?
6. Om du har en anhörig som bär någon form av fallsensor i dagsläget, hur upplevs den?

7. Har du för närvarande någon anhörig du skulle vilja bli informerad i realtid om ifall de råkade ut för en fallolycka?
8. Skulle du alltid vilja se din anhörigas position eller enbart vid en inträffad olycka?
9. Vilken information om olyckan, samt åtgärderna som följer hade du velat kunna se i mobilen/applikationen? (flervalsfråga)
10. Har du några generella tankar, åsikter eller erfarenheter kring fallolyckor och förloppet därefter? (ej obligatorisk)

Svarsalternativen för frågorna var utformade på ett sådant sätt att vid majoriteten av frågor kunde ett eller flera alternativ väljas av de presenterade svaren, exempelvis *Ja*, *Nej* och *Har ingen sensor*. På fråga 4, 9 och 10 fanns möjlighet att skriva en svarstext till frågan. Utöver textrutan med svarstext fanns även flervalsalternativ på fråga 9 som kunde väljas. Några av dessa var *Tidpunkt för olyckan* och *Hur länge personen har legat på marken innan hjälp kommit*.

3.1.4 Användartestning av design

Syftet med utvärderingen var att få inblick i hur väl applikationen fungerar ur användarens perspektiv för att evaluera dess användbarhet samt att undersöka användarnas generella tankar kring applikationens användbarhet. För att uppnå detta har 5 intervjuer genomförts med personer av olika kön och åldrar, med avsikt att utforska deras upplevelser och åsikter. Ett antal frågor har utformats och uppdelats i olika faser.

Den första fasen innehöll följande frågor som rör användarnas tidigare upplevelser med sjukvården efter att en anhörig råkat ut för en olycka.

1. Har en närstående till dig ramlat och gjort sig illa? Om ja: När inträffade det? Kan du berätta om händelseförloppet? Hur och när blev du informerad? Vad kändes viktigast för dig vid tillfället? Vad kunde varit annorlunda för att göra situationen lättare för dig?
2. Har du någon gång varit i kontakt med patientupplysningen? I vilket ärende? Berätta om din upplevelse, gick det fort?
3. Under vilka omständigheter hade en anhörig besökt sjukhuset? Vilka typer av olyckor? Om du tänker på någon speciell, hur långt hade du behövt åka för att ta dig till sjukhuset?
4. På en skala 1 till 5 (1 är inte alls och 5 är väldigt van) hur hade du betygsatt ditt självförtroende på att använda en mobilapplikation?
5. Vad tycker du är viktigt för att en mobilapplikation ska vara värd att använda/ lätt att använda?

Den andra fasen lät intervjupersonerna interagera med design som tagits fram i *Figma*. Detta gav en mer specifik inblick i hur väl designen fungerar genom att utforma ett antal uppgifter som intervjupersonerna ombads fullfölja i applikationsprototypen. Under denna fas togs även mätvärden för att samla in kvantitativ data utifrån olika parametrar. Mätvärden bestod av framgångsbetyg som valdes mellan antingen 0 eller 1 (där 0 motsvarar misslyckande och 1 motsvarar framgång), utöver det noterades tiden för att genomföra uppgiften, antal klick för varje uppgift, nöjdhetsbetyg som valdes mellan 1-5 och även återkoppling från användare vid varje specifik uppgift. Frågorna bestod av:

1. Kan du logga in?
2. Kan du kolla vart Klas Johansson ramlat och vilken gata ambulansen befinner sig på?
3. Kan du navigera till inställningar och ändra textstorlek i applikationen?

4. Kan du skriva ett meddelande till Klas anhöriga?
5. Kan du ringa Klas?
6. Kan du se hur många gånger Klas har ramlat?
7. Kan du hitta dina tidigare meddelanden med Gunnar?
8. Kan du kolla hur processen av händelsen går till och vilken tid ambulansen var på plats?
9. Kan du hitta sammanfattningen av händelsen?
10. Kan du logga ut?

Den tredje fasen innefattade allmänna frågor som avsåg användarnas intryck av designen efter användning. Detta bidrar till att erhålla kvalitativ data om designens användbarhet. Frågorna innehöll:

1. Hur skulle du beskriva din övergripliga upplevelse av applikationen?
2. Vad gillade du mest med applikationen?
3. Vad gillade du minst med applikationen?
4. Var det något som förvånade dig med applikationen?
5. Var det något som du upplevde som frustrerande med applikationen?
6. Finns det något du hade ändrat? Till exempel någon onödigt information och information som saknas.
7. Hade användandet appen kunna underlättas med någon extra förklarande text? Var?

Dessa frågor avslutade användartestningen och utvärderingen av designen.

3.2 Design

Efter att enkätresultaten samlats in påbörjades designprocessen. För att kunna skissa en design behövde en kravspecifikation först sättas samman, som presenterar vad som ska implementeras utifrån resultatet från enkäten och intervjuerna. Kravspecifikationen baserades främst på fråga 9: *Vilken information om olyckan, samt åtgärderna som följer hade du velat kunna se i applikationen?* De fyra alternativ som flest tyckte var viktigast (cirka 80 procent och högre) var:

- Om ambulans är på väg, samt var den i så fall befinner sig.
- Vilket sjukhus och avdelning den skadade befinner sig på.
- Tidpunkt för olyckan.
- Plats för olyckan.

När kravspecifikationen var färdigställd påbörjades designen i *Figma*. Detta inleddes med en väldigt enkel design, se Figur 6.1. Därefter gjordes tre olika designförslag som användes som inspiration för en slutgiltig design. Det första designförslaget, se Figur 6.2, hade en unik komponent vilket var navigationflikarna högst upp på skärmen som flikar på startsidan. Design nummer två som visas i Figur 6.3 var mer blå i färgen. Den sista designen visade förslag på hur applikationen hade kunnat se ut i mörkt läge 6.4. Efter initiala designerna var klara gjordes en slutdesign med hjälp av designförslagen för att det skulle bli så bra som möjligt, se Figur 6.5.

Efter att första iterationen av designen var färdigställd utfördes användartesterna där respons på hur designen fungerade och såg ut samlades in. Då designarbetet är en iterativ process och det är för användarna som systemet byggs gjordes ändringar i designen som användarna önskade.

3.3 Val av Systemarkitektur

För att börja med utvecklingen av applikationen behövde det först bestämmas vilken databashanterare, frontend och backend skulle ska användas. Efter de tidigare nämnda fördelarna med *MongoDB* och tidigare erfarenhet valdes det att användas för att spara data. För frontend valdes ramverket *Expo* som bygger på *React Native* och för detta av samma anledning som tidigare. *Expo* innehåller även många bibliotek som behövs i applikationen såsom *Expo-Notifications* för att notifiera användaren vid fall och *Websocket* för chattfunktionen. *Nativewind* användes för att enkelt utforma (jmf. eng. *style*) de olika komponenterna. *Express.js* valdes som backend då det ger samma programmeringsspråk för både frontend och backend. *Prisma* funkar även med *Express.js* då *MongoDB* kommer användas behövs *Prisma* för att få typer på JSON-datan i databasen.

3.4 Utveckling av backend

I backenden började arbetet med att sätta upp initiala API-slutpunkten "http://localhost:8000/" i index.ts.

```
import express, { Express, Request, Response } from "express";
import dotenv from "dotenv";
import messageRouter from "./src/routes/message";
import userRouter from "./src/routes/user";
import authRouter from "./src/routes/auth";
import chatRouter from "./src/routes/chat";
import relativeRouter from "./src/routes/relative";
import fallRouter from "./src/routes/fall";
import historyRouter from "./src/routes/history";

dotenv.config();
const app: Express = express();
const port = process.env.PORT;

app.use(express.urlencoded({ extended: true }));
app.use(express.json());

app.use("/message", messageRouter);
app.use("/user", userRouter);
app.use("/auth", authRouter);
app.use("/chat", chatRouter);
app.use("/relative", relativeRouter);
app.use("/fall", fallRouter);
app.use("/history", historyRouter);

app.get("/", (req: Request, res: Response) => {
  res.send({ Status: "Still running" });
});

app.listen(port, () => {
  console.log(`      [server]: Server is running at http://localhost:${port}`);
});
```

Listing 3.1: index.ts

Servern lyssnar på förfrågningar på localhost port "8000" som ligger i ".env" filen. Se Figur 6.10 för filstruktur över backenden. När "http://localhost:8000/" kallas kommer servern svara med "Status: "Still Running"". Därefter hanteras resterande *API-anrop* av olika routers, exempelvis *messageRuter* och *userRouter*. Varje router använder sig av *CRUD* (Create, Read, Update, Delete). Routern pratar med en controller som i sin tur skickar vidare förfrågan till en service. I servicen sker all kommunikation

med databasen och resultatet returneras till frontenden. Nedan visas ett exempel på `getMessages` för en chat, detta sköts av `messageRouter`.

```
import express from "express";
const messageRouter = express.Router();
import { messageController } from "../controllers/message.controller";

messageRouter.get("/", messageController.get);

messageRouter.get("/:socialSecurityNumber", messageController.getMessages);

messageRouter.post("/", messageController.create);

messageRouter.put("/:id", messageController.update);

messageRouter.delete("/:id", messageController.remove);

export default messageRouter;
```

Listing 3.2: `message.ts`

```
import { messageService } from "../services/message.service";

const get = async (req: any, res: any) => {
  const socialSecurityNumber = req.body.socialSecurityNumber;

  try {
    res.json(await messageService.get(socialSecurityNumber));
  } catch (err) {
    console.error('Could not get messages', err);
  }
};
```

Listing 3.3: `message.controller.ts`

```
async function get(socialSecurityNumber: string) {
  await prisma.$connect();

  const sensorUser = await prisma.sensorUser.findFirst({
    where: {
      socialSecurityNumber: socialSecurityNumber,
    },
  });

  if (!sensorUser) throw new Error('No sensor user found');

  const chat = await prisma.chat.findFirst({
    where: {
      sensorUserId: sensorUser.id,
    },
  });

  if (!chat) throw new Error('No chat found');

  const messages = await prisma.message.findMany({
    where: {
      chatId: chat.id,
    },
    include: {
      sender: true,
    },
  });
}
```

```

    },
  });

  await prisma.$disconnect();

  const res = messages.map((_message) => {
    return {
      sender: _message.sender.email,
      message: _message.text,
      date: _message.createdAt,
    };
  });

  return res;
}

```

Listing 3.4: message.service.ts

Efter *getMessages* kallas i *BackendConnection.ts* skickas ett *API-kall* till "message" med metoden "get". Detta tar *message.ts* routern hand om och skickar vidare förfrågan till kontrollern. Kontrollern i sin tur tar hand om parametrarna och bodyn i detta fallet *socialSecurityNumber* som indikerar vems chatt meddelandena ska hämtas ifrån. Kontrollern kallar sedan på *get*-metoden i *message.service*. Väl i *message.service.ts* hämtas *SensorUsern* från databasen för att sedan få ut vilken chatt det är med hjälp av *SensorUser.id*. Ifall ingen chat hittas i databasen returneras ett error tillbaka med titeln *No chat found*. Annars hämtas alla messages från databasen och returneras till frontenden. Alla andra *API-kall* hanteras på samma sätt av servern.

3.5 Utveckling av frontend

Utvecklandet började med användargränssnittet och som start skapades en filstruktur för projektet, se Figur 6.9.

Det första som implementerades var *Navigation.tsx* som har i uppgift att navigera mellan olika skärmar med hjälp av *react-navigation*. De första olika skärmarna som skapades var hem- och logga in-sidan. För att göra inloggningen möjlig skapades ett *AuthContext*. Detta är ett kontext som hämtar och sparar login information i *SecureStorage* på telefonen under nyckeln "user". Ifall användaren inte har en giltig *JWT-token* visas loginsidan annars hemskrmen.

```

import React, {
  createContext,
  PropsWithChildren,
  useCallback,
  useContext,
  useMemo,
  useState,
} from "react";
import * as SecureStorage from "expo-secure-store";
import { BackEndConnection } from "../api/BackendConnection";

export interface User {
  email: string;
  token: string;
}

interface UserType {
  user: User | null;
  setUser: (user: User | null) => void;
  login: (email: string, password: string) => void;
}

```

```

    logout: () => void;
    getUser: () => Promise<User | null>;
  }

  export const useAuth = () => {
    return useContext(AuthContext);
  };

  const AuthContext = createContext({} as UserType);

  export const AuthProvider = ({ children }: PropsWithChildren) => {
    const [user, setUser] = useState<User | null>(null);

    const getUser = async (): Promise<User | null> => {
      const _user = await SecureStorage.getItemAsync("user");
      return _user === null ? null : JSON.parse(_user);
    };

    const saveUser = async (_user: User | null) => {
      if (_user) {
        await SecureStorage.setItemAsync("user", JSON.stringify(_user));
      } else {
        await SecureStorage.setItemAsync("user", "");
      }
      setUser(_user);
    };

    const login = useCallback(async (email: string, password: string) => {
      await BackEndConnection.signIn(email, password)
        .then(async (_user: User) => {
          await saveUser(_user);
        })
        .catch((e) => {
          console.log(e);
        });
    }, []);

    const logout = useCallback(async () => {
      await saveUser(null);
    }, []);

    const value = useMemo(
      () => ({
        user,
        setUser: saveUser,
        getUser,
        login,
        logout,
      }),
      [login, logout, user]
    );

    return <AuthContext.Provider value={value}>{children}</AuthContext.Provider>;
  };

```

Listing 3.5: AuthContext.tsx

För att hantera inloggningen använder sig AuthContext av 4 metoder. saveUser och getUser är två metoder som skriver och läser data till SecureStorage på telefonen. Kontextet har även två andra metoder; login och logout. Dessa två använder sig av saveUser och getUser för att sätta User i SecureStorage.

Efter detta gjordes alla komponenter som skulle återanvändas på de olika sidorna. En egen knapp och textfält för att enkelt ändra denna komponent över hela applikationen ifall nödvändigt. När alla komponenterna var klara följdes Figma-designen och resten av sidorna utvecklades. När detta var klart skapades BackendConnection, här skapades först en metod som hette "test". Den skickade ett API-anrop till "http://localhost:8000/" för att se till så att servern fungerade som den skulle. Därefter påbörjades utvecklandet av backenden för att skapa REST-API med alla nödvändiga routes.

```
const makeRequest = async ({
  url,
  method,
  body,
  headers,
}: MakeRequestProps) => {
  const request = "http://192.168.0.189:8000" + "/" + url;
  let requestInit: RequestInit;

  const token = await SecureStore.getItemAsync("user")
    .then((res) => {
      if (res) return JSON.parse(res);
    })
    .then((res: User) => {
      if (res) return res.token;
    });

  if (headers) {
    requestInit = {
      headers,
      method,
      body,
    };
  } else {
    requestInit = {
      headers: {
        Accept: "application/json",
        Authorization: "Bearer " + token,
        "Content-Type": "application/json",
      },
      method: method,
      body,
    };
  }

  const res = await fetch(request, requestInit);

  if (res.status === 200) {
    console.log("successful request");
  }

  const response = await res.json();
  return response;
};
```

Listing 3.6: BackendConnection.ts

För att underlätta förfrågningar till servern skapades en *makeRequest*-metod. Den har två obligatoriska parametrar, "url" och "method", samt två valbara parametrar body och headers.

```
const signIn = async (email: string, password: string): Promise<any> => {
  return await makeRequest({
    url: "auth/signIn",
```

```
method: "post",
body: JSON.stringify({
  email: email,
  password: password,
}),
});
};
```

Listing 3.7: BackendConnection.ts

MakeRequest-metoden skrevs för att återanvända kod i de andra metoderna i BackendConnection klassen. Ett exempel är login metoden där makeRequest körs direkt med url till "auth/signIn". Det är en "post" metod då data skickas till servern, det behövs även skickas med en body med e-post och lösenord för att verifiera inloggningen i backenden.

Kapitel 4

Resultat

4.1 Resultat av intervjuerna

Utifrån de tre intervjuerna erhöles olika resultat. I intervjun med en före detta vårdpersonal inom hemtjänsten nämndes att anhöriga är väldigt lite inblandade i fallolyckor idag. Det finns inget krav för hemtjänsten att kontakta anhöriga vid fallolyckor, men om personen får sådana allvarliga skador att det behövs sjukhusvård måste de kontakta anhöriga och informera att personen är på väg till sjukhus. Fortsättningsvis menas att vårdpersonal ibland kan ringa till anhöriga vid fallolyckor för att vara snälla, men att det inte är deras ansvar att göra det. Det sägs också att det redan är mycket pappersarbete och personalen inte skulle uppskatta att använda en applikation för att kommunicera med anhöriga. Främst för att det tar mycket tid, men också att anhöriga oftast kommer med mycket åsikter och ifrågasätter vårdpersonalen vilket försvårar deras arbete. Utifrån ett presenterat förslag på applikationen anses det som en bra idé så länge det inte försvårar och gör hemtjänstens arbete mer krävande.

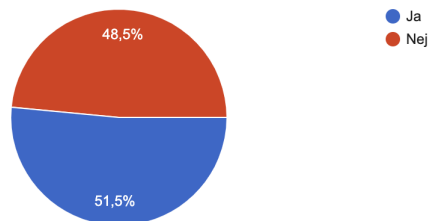
I intervjun med Ulrica Björner och Martin Arvidsson nämner de inledningsvis projektet Snubblometern och redovisar de funktioner som setts vara relevanta för deras projekt, bland annat GPS, händelseförlopp och brukarens kroppsposition. Dagens process när en person ramlar är i dagsläget för lång menar de, och att en sensor med tillhörande system skulle kunna lösa detta problem. De ansåg även att man som anhörig inte vill se alla de värden sensorerna plockar upp i applikationen då de kan skapa oro och stress när man inte själv vet vad datan betyder. Då är lokalisering av personen och händelseförlopp viktigare för anhöriga säger dem.

En ytterligare intervju med en sjuksköterska gjordes som tog upp att alla patienter har fasta, regelbundna hembesök för att hjälpa till med diverse uppgifter. Om en person har ramlat när de inte är där används trygghetstelefonerna för att personen ska kunna tillkalla hjälp. Hon menar också på att personer med sjukdomar såsom demens, kan inte åka ambulans själva efter en fallolycka, vilket gör att man vill att anhöriga ska finnas på plats. När anhöriga dessutom jobbar under dagen blir det svårare för hemtjänst att få tag på dem och meddela om händelsen.

4.2 Resultat av enkäten

Enkäten besvarades av totalt 33 personer (18 kvinnor och 15 män) i åldrar mellan 21 och 77 ($\mu = 44,16$). Drygt hälften var kvinnor (54.5 %) och resterande andel bestod av män (45.5 %). Genom att kvalitativt analysera svaren från enkäten kan slutsatsen dras enligt Figur 4.1 att över hälften av deltagarna (51.5 %) har en närstående som råkat ut för en fallolycka som till följd krävde uppsökande

av vård.



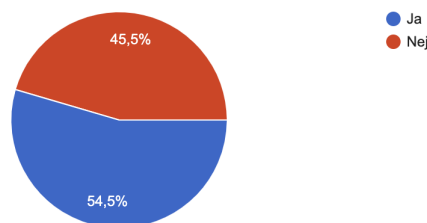
Figur 4.1: Cirkeldiagrammet illustrerar andelen av de anhöriga som efter en fallolycka behövt uppsöka vård.

De flesta av deltagarna (se Figur 4.2) upplevde att de blev dåligt informerade om vårdförloppet och önskar att bli informerade direkt efter fallet.

Dålig information. Dem kunde ha ringt direkt inte 3 dagar senare
Ja, vi visste inte att en fallolycka hade skett och att den anhörige blivit transporterad till sjukhus
Fick samtal från ambulansen. Han har ramlat både ute och inne. Inne har han larm och även hemtjänst flera gånger om dagen. De har då ringt ambulans. Utomhus har främlingar hjälpt till. En app som är kopplat till larmet är egentligen bara den som bor nära och kan rycka ut som har störst nytta av tänker jag. Om jag är 10 mil bort eller fast på jobb kan det bli enorm stress med sån information. Om man då inte får kontakt med någon om hälsotillståndet. När samtalet kom från ambulansen fick jag svar på mina frågor. Men för hemtjänsten borde det vara ett bra hjälpmedel.
Det fanns inget larm utan jag fick reda på det genom mina föräldrar efter att sjukvård hade sökts upp, hade varit bra med ett larm men vet inte hur det skulle kunna ändras i övrigt
Det var mamma som berättade att hon ramlat och slagit sig. Önskar att hemtjänsten kontaktat oss också så vi fått information om vad som hänt, hur illa det var och vilken vård som krävts.
Det ska finnas tydliga journalanteckningar och att jag snarast blir informerad om olyckan.

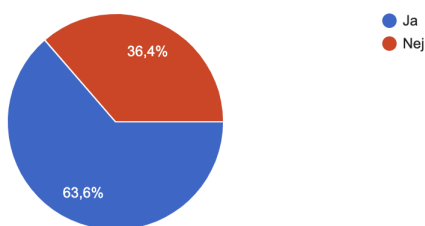
Figur 4.2: Bilden visar deltagarnas upplevelser om vårdprocessen.

Enligt Figur 4.3 har över hälften av deltagarna (54,5 %) för närvarande en anhörig som löper ökad risk för att råka ut för en fallolycka där följden kan vara långvariga skador som kräver sjukvård. Detta antyder att det finns ett intresse bland personer, vilket ligger i linje med det tänkta syftet för applikationen.



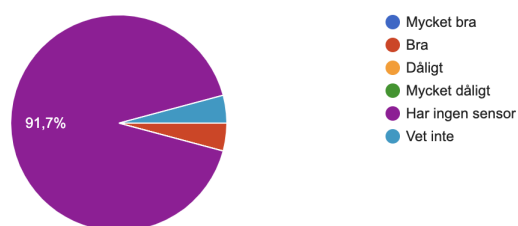
Figur 4.3: Cirkeldiagrammet visar andelen av de anhöriga som har en närstående med ökad risk för att råka ut för en fallolycka.

Många av deltagarna (63.6 %) har för närvarande någon anhörig som de skulle vilja bli informerade i realtid om ifall de råkade ut för en fallolycka, se Figur 4.4. Detta stärker indikationen att det finns en efterfrågan efter en produkt som SeniorSecure ytterligare.



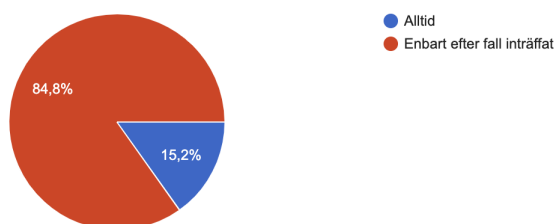
Figur 4.4: Cirkeldiagrammet illustrerar andelen av de anhöriga som önskar bli informerade i realtid vid inträffad olycka.

Utifrån Figur 4.5 kan man utläsa att en stor majoritet (91,7 %) av enkätrespondenterna inte har någon närstående som i dagsläget bär någon form av fallsensor.



Figur 4.5: Cirkeldiagrammet illustrerar hur deltagarna upplever äldre anhörigas fallsensorer i dagsläget.

En stor andel av deltagarna (84.8 %) (se Figur 4.6) skulle föredra att enbart kunna se positionen av sina närstående äldre när en fallolycka inträffat.



Figur 4.6: Cirkeldiagrammet visar andelen anhöriga som önskar se position enbart vid en inträffad fallolycka.

I den avslutande frågan fick respondenterna möjlighet att, utifrån 10 förslag, välja de funktioner som de bedömer är viktigast. Stora andelar av respondenterna, 84.8 % och 81.8 %, skulle vilja bli informerade om huruvida en ambulans är på väg, respektive vilket sjukhus personen befinner sig på. Majoriteten av deltagarna (78.8 % och 87.9 %) vill få information om tidpunkt för händelserna respektive plats

för olyckan. Drygt hälften av deltagarna (57.6 %) vill kunna hitta information om vilka anhöriga som tagit del av information om fallet. Kontaktuppgifter till hemtjänst eller eventuell sjukhusavdelning och hur länge personen har legat på marken innan hjälp kommit ansåg 66.7 % respektive 63.6 % av respondenterna vara viktig information. 42.4 % av de som svarade på enkäten tyckte att eventuella journaluppdateringar skulle vara en viktig funktion. Funktioner som ansågs mindre viktiga var att visa hur många steg den anhöriga hade gått under dagen (12.1 %) och medicinsk information från sensorerna (33.3 %).

4.3 Resultat av användartestning

Resultatet av användartestning gav insikt i de designproblem som finns i prototypen för applikationen, vilka var:

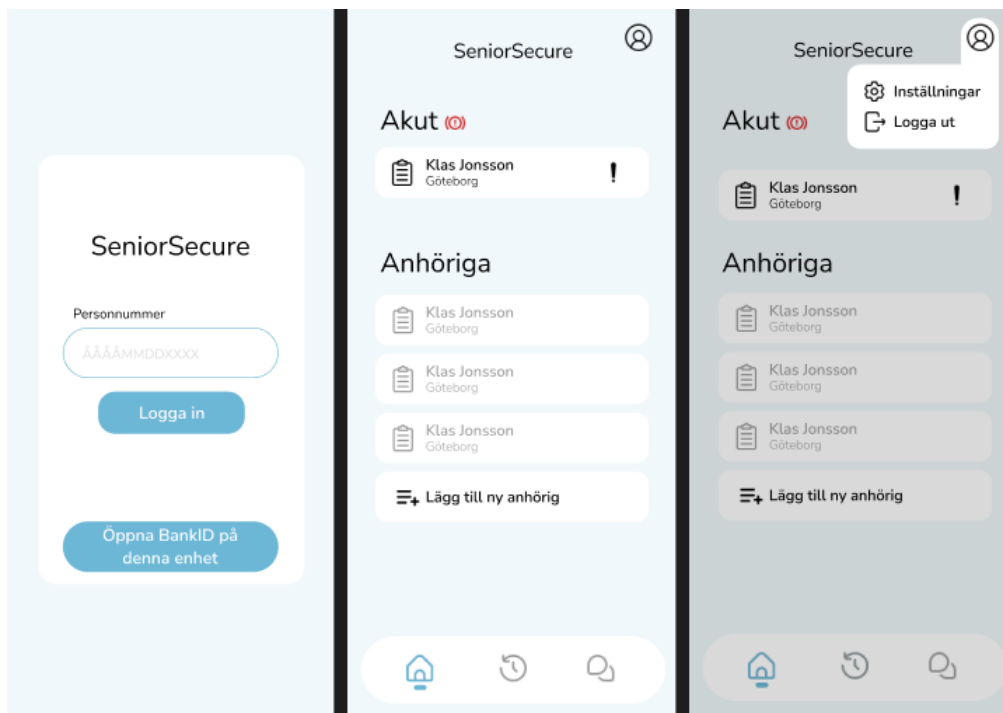
- Ikonerna på kartan är otydliga. Det kan vara bättre att använda bilikon för ambulansen och en annan ikon för sjukhuset, istället för en plusikon (se Figur 4.9).
- Det är oklart vem telefonnumret tillhör (se Figur 4.9).
- Chattrummet med namnet *Klas* är otydligt för att det verkar som att man ska skriva till en person eller en grupp med det namnet. Det kan vara bättre att byta namn på chattrummet för att undvika missförstånd (se Figur 4.10).
- Det blir förvirrande för användaren att det finnas två olika sätt att logga in i applikationen, antingen genom att ange personnummer eller genom att använda BankID (se Figur 4.7).

Analysen av fas 3 beskriver användarnas generella insikt om applikationen, vilka bestod av:

- En förbättring skulle vara att göra de viktiga orden i processfälten fetstilta som *Ramlat*, *Ambulans är på plats* och *Sjukhus* (se Figur 4.9) .
- Det skulle vara bättre att ta bort skrollen i huvudsidan och ha en fryst skärm istället.
- Det råder oklarhet angående logiken i *Tidigare händelser* och *Chatt*. Alla tidigare händelser för en specifik person kan ses under *Tidigare händelser*, fast när man navigerar till *Chatten* visas alla tidigare kommunikationer med de övriga vårdtagande personer som finns i en lista (se Figurer 4.8 och 4.10).
- Lägga till ny funktionalitet i applikationen som tillåter användaren att se vilka mediciner den vårdtagande personen tar.
- Textstorlekar är inte rimliga i vissa delar i gränssnittet, t.ex. storleken på texten *Karta* är större än textstorleken på namnet på den aktuella personen i huvudsidan (se Figur 4.9).
- Att förstå meningen av ikoner som placerades i menyn längs ner till skärmen kan vara utmanande, och en möjlig förbättring kan uppnås genom att definiera ikoner med en tillhörande text under dem (se Figur 4.9).

4.4 Resultat av Design

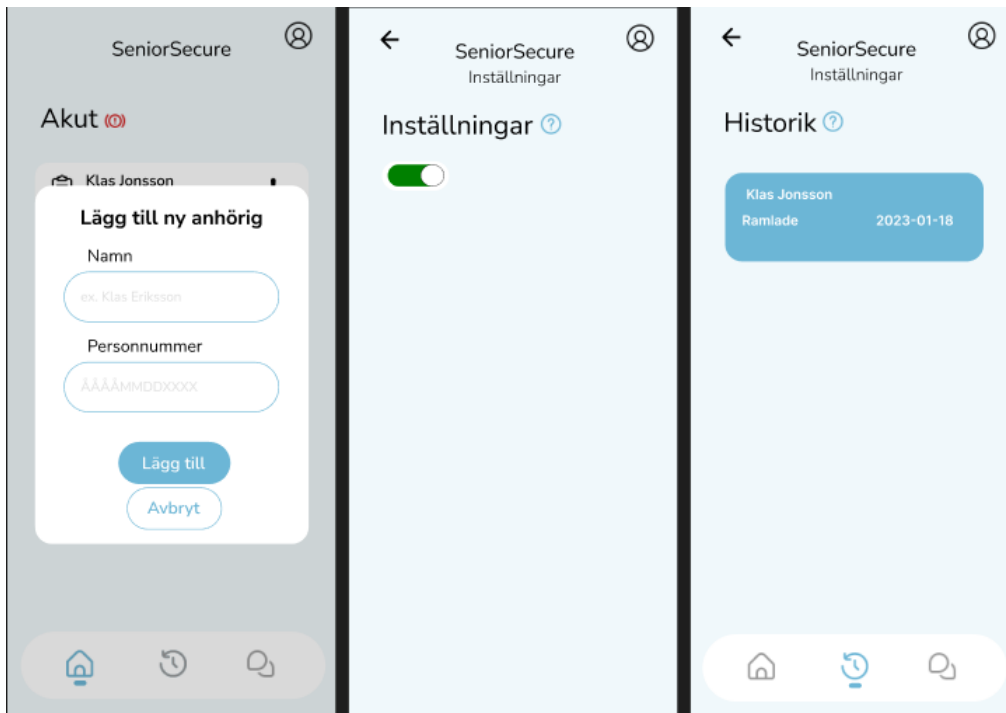
Efter utvärdering och användartestning sammanställdes en slutlig design.



Figur 4.7: Till vänster: Inloggningssidan, i mitten: Startsidan där alla i sekundära målgruppen visas och till höger demonstrerar popup-menyn för att navigera till inställningar och en logga ut-knapp.

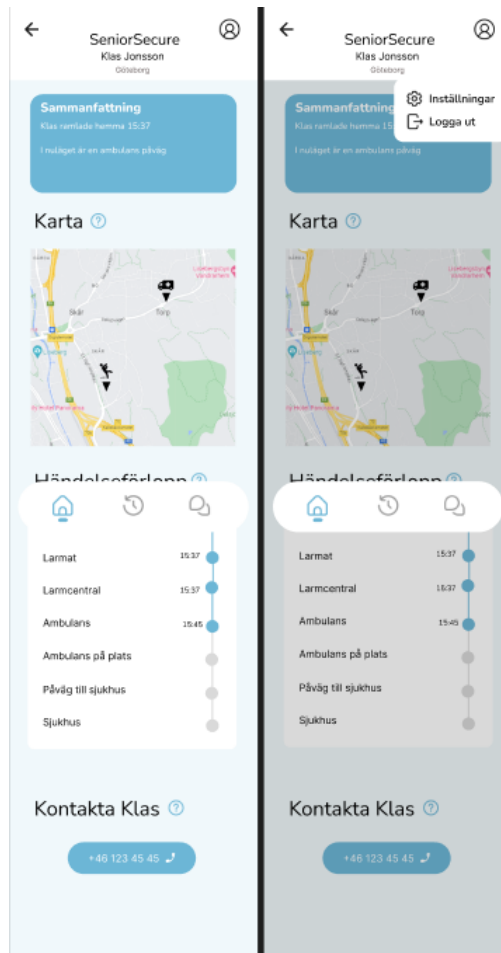
Inloggningssidan (se längst till vänster i Figur 4.7) använder sig av BankID. Ifall BankID-applikationen finns installerad på samma enhet kan den öppnas genom att trycka på den undre knappen 'Öppna BankID på denna enhet'. Ifall BankID inte är installerad på samma enhet används textfältet för att skriva in personnumret och genom att trycka på *Logga in* ska en förfrågan om verifiering skickas till det personnumret.

Startsidan är indelad i två olika listor av anhöriga, en överst för akuta fall vilket betyder att personen har ramlat. Den undre listan, *Anhöriga*, är övriga anhöriga som användaren har lagt till för att bli informerad om vid eventuella fall. Uppe till höger är en profilknapp som vid tryck öppnar en rullgardinsmeny med två olika knappar, *Inställningar* och *Logga ut*. Användaren kan navigera sig runt applikationen med hjälp av en bottenavigering. Den har tre knappar: vänsterknappen navigerar till startsidan, mittenknappen till historiken över tidigare fall och högerknappen för att komma till chattfunktionen.



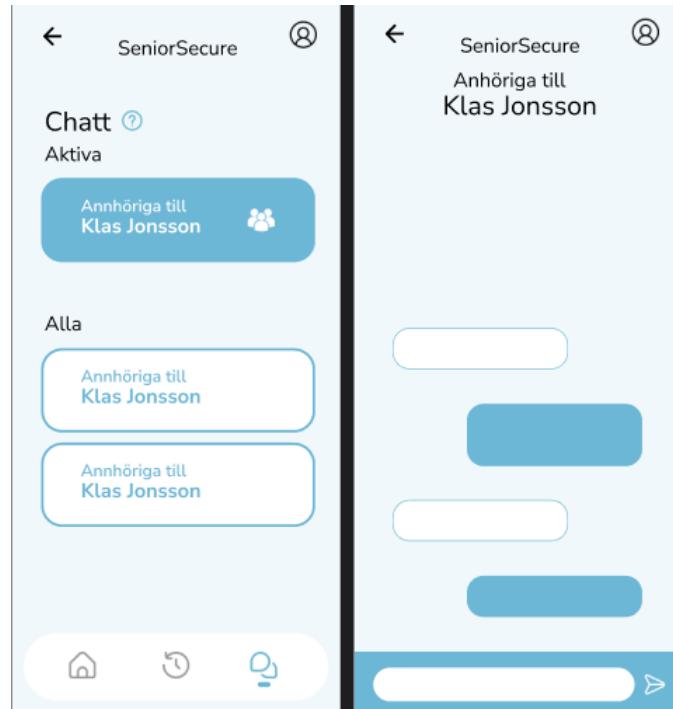
Figur 4.8: Vänstra bilden demonstrerar dialogrutan för att lägga till nya personer från sekundära målgruppen på startsidan, i mitten: inställningarsidan och till höger: historiksidan över tidigare fall.

Vid tryck på *Lägg till anhörig* på startsidan blir man presenterad med en dialogruta, se Figur 4.8. Detta för att rikta användarens fokus extra mycket mot uppgiften. Där skriver man in namn och personnummer på den anhörig som ska läggas till. Bilden i mitten demonstrerar hur inställningarsidan kommer se ut, utöver mörkt läge finns möjligheten att lägga till fler inställningar. Sista sidan visar historiken över tidigare händelser efter användaren tryckt på mittenknappen i botten navigationen.



Figur 4.9: Layout för en i sekundära målgruppen som ramlat. Här visas all information relevant till fallolyckan.

Vid tryck på en anhörig som ramlat under *Akut* på startsidan öppnas en vy där all information om fallet visas, se Figur 4.9. För att göra det lätt för förstagångs användare finns det ett frågetecken över alla olika delar så som karta och händelseförlopp som man kan trycka på för att få en förklaring över vad de olika delarna betyder.



Figur 4.10: Vänstra bilden visar figuren över alla chattarna uppdelad i två listor: aktiva fall och en lista för tidigare chattar. Högra bilden visar layouten för en chatt med tydlig text högst upp för att indikera vilken chatt man är inne på.

Genom att klicka på chattbubblorna nere i menyfältet kommer man till alla aktiva chattar man är medlem i, se Figur 4.10. Efter man tryckt på en specifik chatt öppnas ett enkelt chattforum där man kan skriva med alla andra anhöriga som också är med i chattforumet.

Kapitel 5

Diskussion

I följande avsnitt analyseras och undersöks designprocessen och den resulterande applikationen för att svara på projektets tre frågeställningar. Resultaten diskuteras utifrån den teori som beskrivs i tidigare avsnitt. Efter detta följer en presentation av framtida arbete för applikationens praktiska implementation. Slutsatsen sammanfattar de viktigaste insikterna som uppnåtts under detta projekt.

5.1 Studiens bidrag

Under projektets gång har en prototyp för en applikation som låter anhöriga följa vårdflödet efter en fallolycka i realtid tagits fram där en rad olika funktioner har implementerats och skissats på. Bland annat en chattfunktion för att underlätta kommunikationen mellan anhöriga från den primära målgruppen. Vidare finns en karta och tidslinje för att de ska kunna följa händelseförloppet som följer en fallolycka.

Såvida inte en person från den sekundära målgruppen fallit så illa att ambulans tillkallas finns i dagsläget inga rutiner för hur anhöriga informeras när en fallolycka inträffat. Att ta kontakt med anhöriga efter en olycka kan kännas som en självklar och moraliskt korrekt handling. Trots detta kan det vara en syssla som prioriteras lågt av vård- och akutpersonal på grund av behovet av att snabbt och effektivt hantera och vårda den som fallit. Detta är när en applikation som *SeniorSecure* kan komma till god nytta. Genom att den möjliggör att anhöriga från den primära målgruppen automatiskt förses med information blir en utökad informationsförmedling inte en ytterligare syssla som läggs på den redan tungt belastade vårdpersonalen. Vidare möjliggör applikationen att vårdpersonalen kan koncentrera sig helt på den som fallit.

Att personer från den primära målgruppen vet att anhöriga från den sekundära bär sensorer som kan detektera fall, samt att den primära målgruppen kommer att bli meddelade vid en inträffad olycka kan främja sinnesfrid. Enligt enkätundersökningen som gjordes under projektet framgick att 63.6% av anhöriga skulle vilja bli informerade i realtid när en fallolycka inträffat. Detta kan tolkas som att det finns en efterfrågan bland anhöriga på en lösning där de kan följa händelseförloppet efter incidenten. Det tycks inte heller orimligt att systemet med sensorer och applikation kan bidra till en ökad trygghetskänsla för den sekundära målgruppen. Detta eftersom de vet att anhöriga och vårdpersonal kommer göras medvetna om de skulle råka ut för en fallolycka både i och utanför hemmet. På så sätt kan den sekundära målgruppen potentiellt leva mer självständigt och med större självförtroende, i vetskap om att hjälp finns tillgänglig om det skulle behövas.

För att svara på projektets tredje frågeställning (*Hur skulle den primära och sekundära målgrupperna*

påverkas av en lösning i form av falldetekterande sensorer och tillhörande applikation?) kan det alltså konstateras att en lösning i form av falldetekterande sensorer och tillhörande applikation skulle kunna gynna både den primära och sekundära målgruppen. För den primära målgruppen kan applikationen bidra till en ökad trygghetskänsla och minskad oro. En ökad trygghetskänsla skulle också kunna bli en följd för den sekundära målgruppen utöver ett ökat självförtroende och självständighet.

Applikationens inbyggda chattfunktion skapar vid inträffad fallolycka automatiskt en gruppchatt för alla anhöriga till personen som fallit. Tanken med detta är att underlätta kommunikationen mellan anhöriga, exempelvis vid administration och avstämning vid eventuella sjukhus- och hembesök.

En applikation som *SeniorSecure* möjliggör alltså att anhöriga från den primära målgruppen automatiskt förses med information och kan följa händelseförloppet efter en fallolycka i realtid. Detta kan minska belastningen på vårdpersonalen och öka effektiviteten i vården. Vidare kan chattfunktionen i applikationen underlätta kommunikationen mellan anhöriga och därigenom minska risken för missförstånd och felaktigheter i vården. Genom att involvera den primära målgruppen i vården kan det också skapa en ökad trygghet för personen som drabbats av fallolyckan, vilket kan ha positiv inverkan på rehabiliteringen. Detta besvarar den andra frågeställningen (*Hur kan man med hjälp av sensorer och tillhörande mobilapplikation involvera den primära målgruppen för att uppnå en effektivare vård när en person från den sekundära målgruppen råkat ut för en fallolycka utan att ytterligare belasta vårdpersonal?*).

5.2 Designprocessen

Under designprocessen togs många beslut för att framställa den slutgiltiga designen. Den första designen, se Figur 6.2, hade en unik design med flikar högst upp på mobilen för att navigera sig. Detta röstades om och bestämdes att det var lite för innovativt för att alla ska kunna förstå den funktionen. I design 2, se Figur 6.3, tyckte alla medlemmar i projektet hade en snygg färg, men att den blå nyansen här är lite för stark. Så det bestämdes att använda sig av en ljusare blå som primär färg i arbetet. I design 3, se Figur 6.4, var listan för att lägga till anhöriga en smidig funktion som beslutades att den skulle vara med i slutgiltiga designen. Att använda sig av mörkt läge var dock något som riktade sig mot den yngre delen av målgruppen och något som kan vara irriterande för de äldre i målgruppen. Mörkt läge bestämdes då att det ska finnas som en inställning man kan sätta på och av.

5.3 Gränssnittsdesign

Stort fokus har legat på att ta fram ett gränssnitt för en applikation som är välanpassad till den primära målgruppen. Följande besvarar den första frågeställningen: *Hur utformas en mobilapplikation med syfte att informera den primära målgruppen om att en anhörig från den sekundära målgruppen har fallit?*

En tidig insikt under projektet var att eftersom den sekundära målgruppen har gemensamt att de befinner sig i riskgrupp, finns stort utrymme för variation av andra karaktärsdrag, så som ålder, utbildning och socioekonomisk status. Det gör att den primära målgruppen kan förväntas variera ännu mer eftersom den gemensamma faktorn här är att de är närstående till någon som tillhör riskgrupp. Detta ställer krav på att gränssnittets inklusivitet och användbarhet.

Det slutliga gränssnittet förser användaren med kognitivt stöd genom att exempelvis förse denne med möjligheten att enkelt försöka ta kontakt med den anhöriga som har ramlat (se Figur 4.9). Denna funktion kommer förmodligen vara till störst nytta vid den lindrigare typen av fallolyckor där den

anhöriga från den sekundära målgruppen klarat sig så pass väl att de kan svara på samtalet. Ett annat exempel på kognitivt stöd är appens inbyggda chattfunktion (se Figur 4.10), som automatiskt skapar en chatt för alla anhöriga kopplade till den som fallit. På så sätt uppstår ingen förvirring om vem som ska skapa chatten, inte heller finns risken att någon glöms bort. Här kan sedan kommunikationen enkelt äga rum i ett format som är välkänt från andra chattforum och applikationer.

Under designprocessen lades stor vikt vid att hålla gränssnittet så minimalistiskt och simpelt som möjligt eftersom detta förser användaren med informationsstöd. En förutsättning för en enkel design är att ingen onödig information förekommer i gränssnittet. Exempelvis är det första användaren ser när den kommer in i fall-vyn, en sammanfattning som talar om vad som inträffat, samt var och när, dessutom visas även den aktuella statusen i förloppet här. Detta är sedan följt av en karta där man kan få en bättre bild över hur exempelvis vårdpersonalen ligger till. Därefter får man en mer detaljerat överblick över händelseförloppet. Med stöd i enkäten valdes annan hälsodata såsom puls och blodtryck att uteslutas från denna vy, eftersom den primära målgruppen inte kan förväntas tolka och förstå den. Vyn visar därmed endast sådant som är av värde för den primära målgruppen och hjälper dem att bilda sig en uppfattning av läget.

En princip som är nära kopplad till begreppet användbarhet är universell design, vilket innebär att en produkt ska vara utformad så att den kan användas av en så bred och varierad målgrupp som möjligt. Återigen är den enkla och avskalade designen ett verktyg för att göra applikationen tilltalande för fler. Eftersom onödig funktionalitet och information är begränsad minskar risken för att användaren behöver leta för att hitta specifik information, även om användaren har begränsad erfarenhet av applikationen. Utöver det försågs designen även med info-knappar (se Figur 6.7). Syftet med dessa är att användaren enkelt ska kunna få förklarat för sig hur innehållet på sidan ska tolkas eller användas. Dessa knappar finns i applikationen, men på grund av tidsbrist har inga slutgiltiga förklarings-texter komponerats. Slutligen har applikationen gjorts responsiv, alltså att elementen i gränssnittet anpassar sig efter storleken på den aktuella skärmen. Detta gör att användarens upplevelse inte kommer bero på vilken typ av telefon de har.

Applikationens primära syfte är som sagt att förse användaren med uppdateringar och information. En följd av detta är att de tillfällen användaren riskerar att begå misstag med omfattande konsekvenser är relativt få. Majoritet av inputen applikationen tar emot från användaren är knapptryck med syfte att navigera användaren i applikationen. För att det ska vara enkelt att gå tillbaka om man råkar trycka fel försågs gränssnittet med bakåtpilar (se Figur 4.10). Med vetskap om detta kan användaren med större självförtroende pröva sig fram när de letar efter någon specifik funktion. Dessutom kan användaren känna sig tryggare genom navigeringsfältet som är placerat längst ned i gränssnittet vilken blir ett ytterligare sätt att gå tillbaka (se Figur 4.10).

Det är med andra ord i en begränsad utsträckning som användaren förser applikationen med input i annan form än navigerande knapptryckningar. Ett av de fåtal exempel på tillfällen där användaren kan begå misstag är chattfunktionen, där användaren exempelvis kan råka skicka ett meddelande i fel grupp, eller råka skicka ett icke färdigskrivet meddelande. Av denna anledning bör det finnas en funktion för att radera meddelanden. Detta har dock inte hunnits implementeras inom ramarna för projektet.

Under designprocessen har bland annat Normans sju designprinciper funnits i åtanke. Trots att dessa principer riktar sig till design av vardagliga ting generellt har de med fördel kunnat tillämpas i gränssnittsdesignen. En av principerna säger att kunskap från både verkliga livet och minnet ska användas. Detta har beaktats bland annat vid valet av ikoner för applikationen för att se till att den aktuella funktionen tydligt speglar en välkänd funktionalitet från minnet eller verkliga livet tillsammans med förklarande texter där det kan finnas tveksamheter, i enlighet med principen.

Applikationens generella struktur, men även struktur av uppgifter är skapade med inspiration från Normans principer om att förenkla strukturen av uppgifter och principen om att strukturera rätt. På så sätt att uppgifterna som kan utföras i appen systematiskt hålls så lika varandra som möjligt och att relationerna mellan intentioner och möjliga handlingar tydliggörs som exempelvis att vid alla tillfällen där man kan backa tillbaka används samma bakåtpil. Även principen om att göra saker synliga tillämpas exempelvis på så sätt att ikonerna tydligt ska avslöja vilka effekterna blir om man klickar på dem. Enligt Norman bör designern alltid anta att användaren kommer att göra fel och därför ska det med enkelhet gå att ångra handlingar vilket det i applikationen går att göra genom att klicka på den återkommande bakåtpil-ikonen. Tillbakapilen är även ett exempel på implementation av tillvänjning, genom att möjliggöra för användaren att återgå till föregående sida. Dock kan den också vilseleda användaren om manövreringar används i en situation där de inte fungerar som användaren förväntade sig, vilket kan leda till att användaren fokuserar på processen. Applikationen använder även *Bottennavigering* för att förse användarna med ett välbekant sätt att navigera i applikation.

5.4 Användarfeedback och tester

Efter att ha erhållit resultaten från enkäten, inleddes en analys av dem. Slutsatsen från analysen indikerar att applikationen kommer kunna uppfylla de specificerade behoven hos den primära målgruppen genom att omedelbart notifiera anhöriga när en händelse inträffar och sedan tillåta dem att kontinuerligt följa händelseförloppet. Mobilapplikationen utvecklades med en användarcentrerad designmetodik, därmed har det eftersträövats att innehållet och funktionerna stämmer överens med användarnas prioriteringar och värderingar, vilket talar för att den uppfyller användarnas behov.

Resultatet av användartestningen gav en mer grundlig förståelse för de designproblem som fanns i prototypen. Syftet med fas 1 var att få insikt om applikationsbehoven genom relevanta frågor. I fas 2 låg fokus på användartestning, och genom att analysera svaren fick vi feedback angående de brister som fanns i designen. Analysen av fas 3 tolkade användarnas generella feedback om applikationen. Efter att ha sammanställt och analyserat användarfeedbacken kan man konstatera att applikationen kommer att kunna uppfylla de behoven hos den primära målgruppen och att den utvecklats med en användarcentrerad designmetodik som prioriterar användarnas preferenser.

5.5 Etiska aspekter

En av avgränsningarna för projektet var att ingen hänsyn tas till rådande lagar och begränsningar gällande integritet och informationsdelning. Detta är något som behöver ses över. Systemet bakom applikationen kommer samla in och lagra data om den sekundära målgruppens dagliga rutiner, aktiviteter och hälsodata. Dessa kan potentiellt användas för andra ändamål än falldetektering. Det är viktigt att överväga hur denna data kommer användas och vem som kommer ha tillgång till den och äga den, samt se till att den sekundära målgruppens integritet respekteras.

Det finns en ytterligare etisk aspekt som behöver beaktas, nämligen det faktum att personer från den sekundära målgruppen kanske inte helt förstår innebörden av att använda en falldetekterande sensor med en applikation som *SeniorSecure*. Det behöver dels vara tydligt vilken information som samlas in av det bakomliggande systemet och dels vilken informationen anhöriga från den primära målgruppen kommer ha tillgång till. Med andra ord är det viktigt att se till att den sekundära målgruppen kan ge ett informerat samtycke till att använda tekniken och att de förstår hur deras data kommer att användas.

Vidare bör det även ses till att tillgången till applikationen inte är begränsad av faktorer som

socioekonomisk status, geografisk plats eller digital kompetens. Det är viktigt att överväga hur man säkerställer att applikationen är tillgänglig för alla äldre individer som skulle kunna dra nytta av den, oavsett dessa faktorer. Ett sätt att åstadkomma detta är att låta det ingå i det kommunala utbudet av hjälpmedel.

5.6 Tekniska utmaningar

Första tekniska utmaningen var typanDET av databasen med *Prisma* då *MongoDB* valdes att användas. Först valdes att skapa en *User*-modell där *User* kunde ha *Relatives*, en array av primära målgruppen i form av *User*. Då det anades att många attribut skulle vara samma i databasen etablerades en själv-relation. Men när en *Fall*-modell och *Historik*-modell skulle implementeras blev det tydligt att användare från den sekundära målgruppen även behöver ha en relation till fall, medan användare från den primära målgruppen bör ha en relation med historik. Eftersom datan sparas i en *noSQL*-databas är det inga problem att spara tomma attribut. Men med så många valbara attribut blev det mer logiskt att skapa två separata modeller, en för *User* (primära målgruppen) och en för *SensorUsers* (sekundära målgruppen).

Implementationen av *BankID* gick så långt att man kunde öppna *BankID* i appen genom att trycka på 'Öppna BankID på denna enhet' på inloggningsskärmen. Men när förfrågningar om verifikation skulle skickas behövde man en nyckel av BankID. För att få nyckeln behövdes ett företagsnummer, men då ett företag inte var skapat bestämdes att istället använda sig av e-post och lösenordsinloggning.

Profilikonen uppe till höger på sidan ställde till problem då rullgardinsmenyn skulle vara en pop-upp meny, det blev snabbt tydligt att komponenter från sidhuvudet (jmf. eng. *header*) lade sig under huvudinnehållet (jmf. eng. *body*). Detta gjorde att man inte såg *Inställningar*- och *Logga ut*-knapparna i menyn. Istället fick en dialogruta användas som lade sig över hela sidan. För att det skulle se så bra ut som möjligt gjordes den liten och positionerades uppe till höger.

5.7 Begränsningar

På grund utav arbetets utformning tillkom även vissa begränsningar i projektet. Dessa begränsningar bör ha i åtanke för framtida arbeten för att åstadkomma ett resultat med så få felkällor som möjligt.

Att använda en enkät som datainsamlingsmetod har både för- och nackdelar. Metoden möjliggör insamling av en stor mängd data utan att vara särskilt tidskrävande för deltagarna såsom till exempel intervjuer eller samtal. Däremot kan faser för dataanalys vara mer krävande då det riskerar att finnas felkällor att ta hänsyn till. En viktig aspekt att ha i åtanke är hur frågorna i enkäten ställs och formuleras, kan de uppfattas ledande eller partiska? Ett sätt att göra frågorna mindre ledande kan vara att bara låta respondenterna svara i fritext istället för att erbjuda svarsalternativ, risken med detta är dock responströtthet vilket kan bidra till felaktiga eller ofullständiga uppgifter. Vidare kan det vara svårt att ta fram ett undersökningsurval som är representativt för populationen som studeras beroende på faktorer som tekniska färdigheter hos enskilda individer och vilka som nås av enkäten, vilket bör beaktas vid generalisering av resultatet. Även djupet av information kan vara begränsat i jämförelse med andra datainsamlingsmetoder, därför bör resultatet från en enkät kompletteras med undersökningar som samtal och intervjuer för att fånga mer rik och kvalitativ data och förstå studiedeltagarens erfarenheter och perspektiv. För uppnå ett mer tillförlitligt resultat för eventuell framtida arbete skulle en större enkät kunna användas för att få en tydligare bild av mer komplex data.

5.8 Framtida arbete

För att applikationen ska kunna bli konventionell finns ett antal aspekter som bör ses över så som lagstiftning och utveckling av övriga system. I dagsläget har de inblandade parterna i vårdkedjan separata system som är helt isolerade från varandra. Detta är ett hinder som kraftigt begränsar omfattningen av den automatiserade händelserapportering som har en väsentlig roll i applikationen. Därmed är ett viktigt steg att se över rådande lagstiftning angående integritetsrätt. I denna process behöver den sekundära målgruppens integritetsrätt balanseras med den primära målgruppens önskan att hålla sig informerade och involverade i deras vård. Att antalet äldre individer ökar i samhället antyder även att den sekundära målgruppen växer. Med andra ord kan det finnas en växande efterfrågan bland allmänheten på en lösning som möjliggör för nära anhöriga att hålla sig bättre informerade och delaktiga i vården av sina nära och kära. Lagstiftning som begränsar informationsutbytet efter en olycka skulle därför ses som något föråldrat eller i otakt med samtidens värderingar kring vikten av familjens engagemang i vården.

Det finns potential att implementera applikationen i det vårdssystem som existerar idag. I samband med att en sensor detekterar ett fall och skickar detta till larmcentralen, skickas information om olyckan även till *SeniorSecure*-servern och de anhöriga. För att de sedan ska kunna hållas uppdaterade förloppet som följer, krävs att servern kan kommunicera och ta emot information från ambulans, sjukhus eller andra aktörer. I detta avseende spelar standardiserade dataformat, såsom *FHIR*, en viktig roll. Standarden gör att servern vet hur datan kommer vara formaterad och strukturerad, oavsett vem avsändaren är. Därav kan samtliga statusuppdateringar hanteras på samma sätt av systemet, det finns alltså inte något behov att skraddarsy kommunikationen mellan *SeniorSecure* och varje enskild avsändare. Med andra ord finns möjligheten att göra servern till en plugin som enkelt kan kopplas till andra system och utbyta information med dem.

Förutsatt att övriga systemet med sensorer och automatisk händelserapportering fungerar väl finns ändå ett behov för en motsvarande applikation för den sekundära målgruppen. I prototypen kan vem som helst lägga till en anhörig så länge vederbörande har personens personnummer. I en slutgiltig version av applikationen bör det krävas godkännande från personen från den sekundära målgruppen för att kunna ta del av personens hälso- och händelseinformation. Med andra ord skulle ett koncept likt vänförfrågningar tas fram. Det är dock viktigt att den sekundära målgruppen inte av misstag godkänner förfrågningar eftersom detta skulle resultera i att känslig information når fel personer. Med det sagt bör det tittas närmare på hur förfrågningarna ska utformas för att förebygga detta.

Att den sekundära målgruppen har en applikation som motsvarar *SeniorSecure* öppnar även upp för att inkludera dem i gruppchattarna som skapas vid inträffade fallolyckor. Detta underlättar kommunikationen mellan de två målgrupperna vilket återigen gör det lättare att exempelvis stämma av och planera besök. Att inkludera individerna från den sekundära målgruppen i gruppchatten kan få dem att känna sig mer delaktiga och inkluderade. Det kan också bidra till en ökad känsla av kontroll genom att de tillåts uttrycka preferenser och bekymmer direkt till sina anhöriga (den primära målgruppen). Å andra sidan kan en begränsad teknisk vana och erfarenhet göra att den sekundära målgruppen upplever svårigheter och till och med frustration i samband med gruppchatten. Utöver detta kan det finnas en större risk för felkommunikation eller missförstånd om den äldre individen har svårt att kommunicera eller förstå meddelanden i chatten. I slutändan kommer beslutet att inkludera den sekundära målgruppen i chattfunktionen bero på faktorer såsom deras tekniska skicklighet och självförtroende, deras preferenser och behov, samt den primära målgruppens preferenser. För att navigera detta kan det vara till hjälp att genomföra användartester och samla in feedback från alla parter som är involverade i vårdprocessen för att informera om detta beslut.

Ett ytterligare potentiellt användningsområde för chattfunktionen är att den skulle kunna användas

för att sköta kommunikationen mellan vårdpersonal och anhöriga. De anhöriga skulle kunna ställa frågor till exempelvis läkare, vårdpersonal, sjuksköterskor. En möjlighet är att de skulle kunna förbereda standard svar för de vanligast förekommande frågorna. Innan detta implementeras är det dock viktigt att det står klart att detta minskar personalens arbetsbörda. En potentiell risk är att en mer effektiv kommunikation skulle leda till att fler frågor ställs, vilket i sin tur skulle kunna leda till en större arbetsbörda än innan. En ytterligare nackdel är att det kan bli en klumpigare konversation. Chattformatet är inte lika effektivt när det kommer till att göra förtydliganden och att ställa följdfrågor. Detta skulle kunna leda till frustration hos både vårdpersonal och anhöriga.

I framtida arbete bör det även ses över hur applikationens användningsområde kan utökas. Exempelvis hade den kunnat inkludera andra typer av olyckor och incidenter som exempelvis är kopplade till hjärtat eller en eventuell diabetes. Dessutom finns möjligheten att bredda den sekundära målgruppen så att den inte enbart innefattar personer som ingår i en riskgrupp.

Avslutningsvis är det också viktigt att undersöka och identifiera de fall där en lösning som *SeniorSecure* inte fungerar tillräckligt bra. Ett av intervjuobjekten nämnde att exempelvis personer med demens inte tillåts åka ensamma i ambulans. Helst ska en anhörig följa med, men om de inte går att få tag på landar det ansvaret på vårdpersonalen. I nuläget har inte *SeniorSecure* utvecklats för att kunna ersätta den telefonkontakt som äger rum mellan vårdpersonal och anhöriga. Det behöver finnas ett sätt för anhöriga att bekräfta huruvida de kan bistå eller ej, något som inte är möjligt med den aktuella händelserapporteringen i applikationen. Detta är ett område inom vilken *SeniorSecure* behöver utvecklas mer, det behöver även utredas om det finns fler grupper inom den sekundära målgruppen som kräver speciella åtgärder och hantering.

Kapitel 6

Slutsats

Under projektet har ett 'proof of concept' för en applikation som låter anhöriga följa vårdflödet efter en fallolycka i realtid tagits fram. Applikationen *Senior Secure* är uppbyggd av en frontend och backend, där frontenden är skriven i *Expo* ett ramverk till *React Native* och backenden i *Express.js* ett ramverk för *Node.js*. Den har potential att bidra till att förbättra vårdkedjan genom att automatiskt informera anhöriga från den primära målgruppen om olyckor och därigenom tillåta vårdpersonalen att enbart koncentrera sig på patienten. Den enkätundersökning som gjordes tidigt under projektet antyder att det finns efterfrågan på en sådan lösning. Applikationen kan också bidra till en ökad trygghetskänsla för den sekundära målgruppen genom att se till att hjälp finns tillgänglig om det skulle behövas. Applikationens inbyggda chattfunktion underlättar därutöver kommunikationen mellan anhöriga vid administration och avstämning vid eventuella sjukhus- och hembesök.

För att försöka uppnå en mer intuitiv och lättanvänd applikation som bättre tillgodoser användarnas behov och ökar chansen att applikationen används på ett effektivt sätt lades stor vikt upp utformandet av själva gränssnittet. En svårighet med applikationen är att den ska vara användbar för en grupp individer med stor variation beträffande exempelvis ålder och mobilvana. Utifrån litteraturstudien drogs slutsatsen att ett sätt att värna om en bred målgrupp är att se till att designen är så simpel som möjligt och inte innehålla någon onödig information eller funktionalitet.

Innan applikationen är redo att användas i praktiken kvarstår dock en rad hinder att ta itu med. Många av dessa hinder är etiska överväganden som behöver göras. Utöver det kan fler och större enkätundersökningar exempelvis ge viktiga insikter om variationer i vårdkedjan och användarkrav i landet.

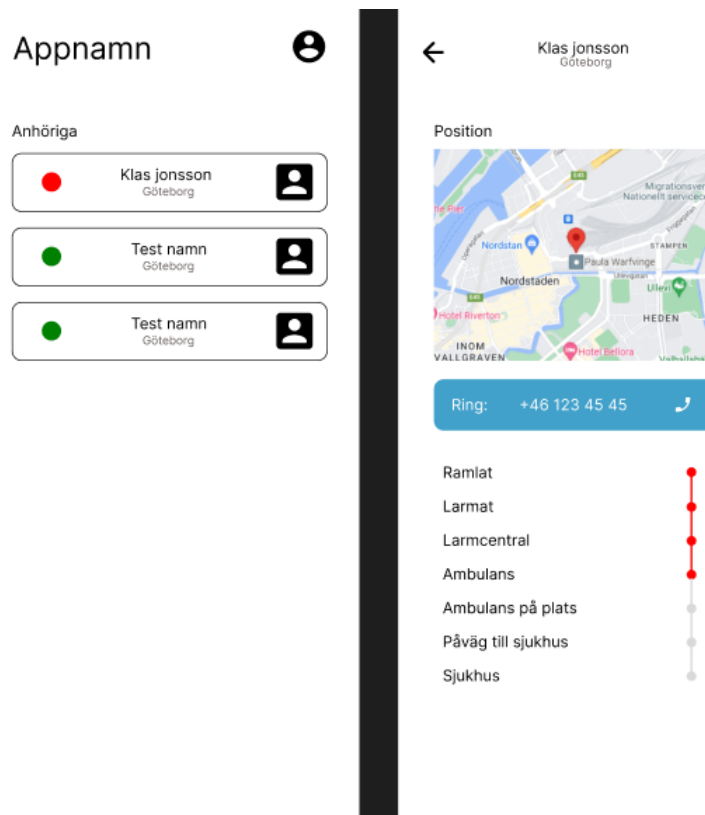
Projektet har påvisat potentialen i att använda teknologi för att förbättra vårdkedjan och öka trygghetskänslan hos både den primära och sekundära målgruppen. En viktig faktor för att uppnå detta är att designa en användarvänlig applikation med ett intuitivt gränssnitt. Projektet har också identifierat flera etiska överväganden som behöver hanteras innan applikationen kan implementeras i praktiken. Fortsatta enkätundersökningar kan också ge värdefull information om variationer i vårdkedjan och användarkrav i landet. Genom att ta itu med dessa hinder kan projektet möjliggöra en förbättrad och mer effektiv vårdkedja för patienter och anhöriga.

Litteratur

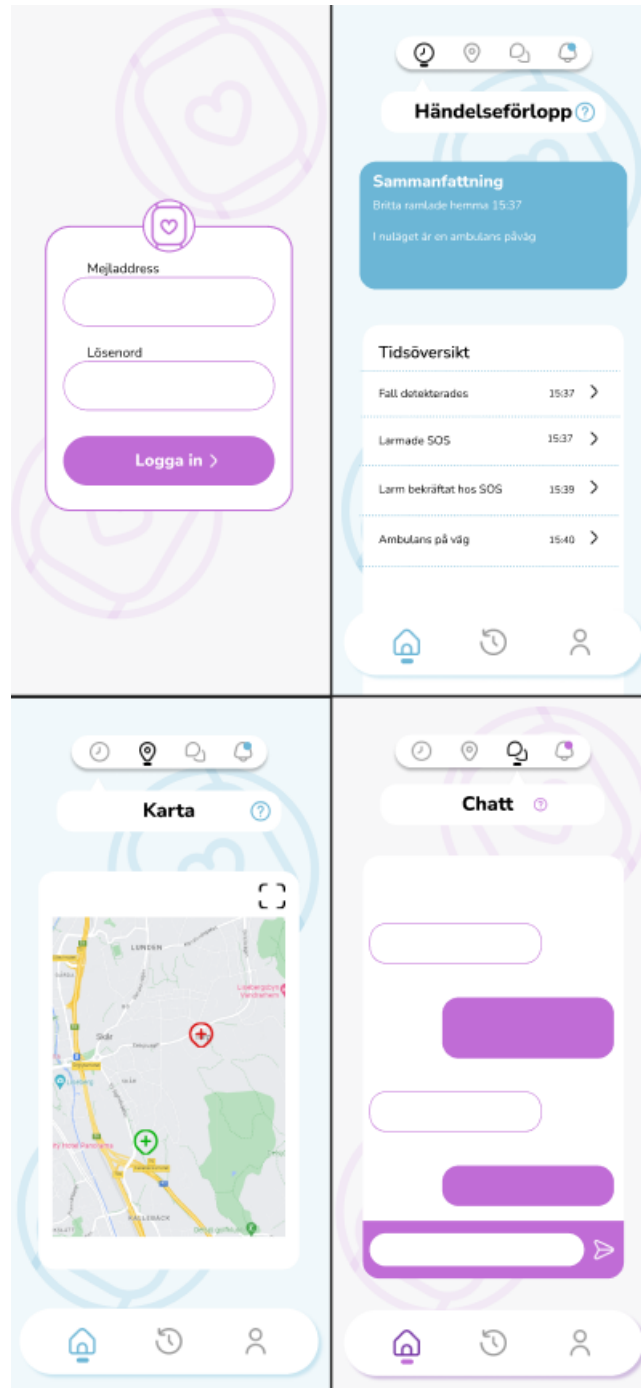
- [1] Folkhälsomyndigheten. "Fallolyckor bland äldre, vårdade." (2022), Tillgänglig: <https://www.folkhalsomyndigheten.se/fu-fallolyckor-bland-aldre>. (Hämtad: 2023-02-03).
- [2] R. Stockholm. "Fallolyckor." (2019), Tillgänglig: <https://www.folkhalsorapportstockholm.se/rapporten/sjukdomar-och-skador/fallolyckor/>. (Hämtad: 2023-02-03).
- [3] M. B. och Zdeněk Berka, "Falls: a significant cause of morbidity and mortality in elderly people," *Vnitr Lek.*, vol. 64, dec. 2018. doi: 10.36290/vn1.2018.154.
- [4] G. C. A. och Shou Lin Low och Choon How How, "Approach to falls among the elderly in the community," *Singapore Med J.*, vol. 61, mars 2020. doi: 10.11622/smedj.2020029.
- [5] S. FN-förbundet. "Globala målen för hållbar utveckling." (2021), Tillgänglig: <https://fn.se/globala-malen-for-hallbar-utveckling/>. (Hämtad: 2023-03-02).
- [6] M. K Karlsson, T. Vonschelow och B. E Rosengen, "Prevention of falls in the elderly: A review.," *Scandinavian Journal of Public Health.*, vol. 41, nr 5, 2012. doi: 10.1177/1403494813483215.
- [7] T. E. et. al. "Fallprevention - en kostnadseffektiv åtgärd?" (2022), Tillgänglig: <https://www.socialstyrelsen.se/globalassets/sharepoint-dokument/artikelkatalog/ovrigt/2022-5-7923.pdf>. (Hämtad: 2023-02-03).
- [8] G. Stad. "Så fungerar trygghetstelefon." (n.d), Tillgänglig: <https://goteborg.se/wps/portal/start/aldre/hjalp-i-hemmet/trygghetslarm/trygghetslarm/sa-fungerar-trygghetslarm>. (Hämtad: 2023-03-03).
- [9] U. B. och Martin Arvidsson, privat kommunikation, mars 2023.
- [10] U. Björner, privat kommunikation, mars 2023.
- [11] S. U. et. al., "Latest Research Trends in Fall Detection and Prevention Using Machine Learning: A Systematic Review," *Sensors (basel)*, vol. 21, aug. 2021. doi: 10.3390/s21155134.
- [12] Apple. "Använda Falldetektering med en Apple Watch." (2023), Tillgänglig: <https://support.apple.com/sv-se/HT208944>. (Hämtad: 2023-04-25).
- [13] W. e. a. Sasaki, "Estimating Symptoms Caused by CIPN Using Mobile and Wearable Devices," 2020. doi: 10.1145/3410530.3414435.
- [14] A. A. Smith, R. Li och Z. T. H. Tse, "Reshaping healthcare with wearable biosensors," *Scientific Reports*, vol. 13, nr 1, 2023. doi: 10.1038/s41598-022-26951-z.
- [15] Socialstyrelsen. "Anhöriga som vårdar eller stödjer närstående äldre personer." (dec. 2020), Tillgänglig: <https://www.socialstyrelsen.se/globalassets/sharepoint-dokument/artikelkatalog/ovrigt/2020-11-7045.pdf>. (Hämtad: 2023-04-28).
- [16] Patientdatalagen, SFS 2023:167, Socialdepartementet, Stockholm, Sverige: Regeringskansliet, maj 2008. Tillgänglig: https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/patientdatalag-2008355_sfs-2008-355.

- [17] Socialtjänstlagen, SFS 2022:1034, Socialdepartementet, Stockholm, Sverige: Regeringskansliet, juni 2001. Tillgänglig: https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/socialtjanstlag-2001453_sfs-2001-453.
- [18] A. Williams, "User-Centered Design, Activity-Centered Design, and Goal-Directed Design: A Review of Three Methods for Designing Web Applications," i *Proceedings of the 27th ACM International Conference on Design of Communication*, ser. SIGDOC '09, Bloomington, Indiana, USA: Association for Computing Machinery, 2009, s. 1–8. DOI: 10.1145/1621995.1621997.
- [19] Z. H. och Morad Benyoucef, "A systematic literature review of mobile application usability: addressing the design perspective," 2022. DOI: 10.1109/ICoDSE56892.2022.9972176.
- [20] A. Parlakkiliç, "Evaluating the effects of responsive design on the usability of academic websites in the pandemic," *Education and Information Technologies*, vol. 27, nr 1, 2021. doi: 10.1007/s10639-021-10650-9.
- [21] Y. G. J. et. al., "A usability checklist for the usability evaluation of mobile phone user interface," *International Journal of Human-Computer Interaction*, vol. 20, mars 2006. doi: 10.1207/s15327590ijhc2003_3.
- [22] M. F. Story, "Principles of universal design," *Universal design handbook*, vol. 2, 2001.
- [23] J. e. a. Tidwell, *Designing Interfaces: Patterns for Effective Interaction Design*. O'Reilly Media, 2020. Tillgänglig: <https://ebookcentral.proquest.com/lib/chalmers/detail.action?docID=5996435>.
- [24] M. R. J. Estuar, M. De Leon, M. D. Santos, J. O. Ilagan och B. A. May, "Validating UI through UX in the Context of a Mobile - Web Crowdsourcing Disaster Management Application," s. 1–4, 2014. doi: 10.1109/ICITCS.2014.7021823.
- [25] C. Rankin, T. Abrams, S. R. Barry m. fl., "An updated and revised description of the behavioral characteristics of habituation," *ScienceDirect*, vol. 92, aug. 2009. doi: 10.1016/j.nlm-022-02890-2.
- [26] D. A. Norman, *The Design of Everyday Things*. The Persus Books Group, 2002.
- [27] I. Silva, P. Cardoso och B. Giesteira, *Strategies of Intentional Friction in the User Interface of Digital Games* (Springer Series in Design and Innovation). 2023, vol. 27. DOI: 10.1007/978-3-031-20364-0_5.
- [28] B. D. S. et. al., "The great typography bake-off: comparing legibility at-a-glance," *Ergonomics*, vol. 63, april 2020. doi: 10.1080/00140139.2020.1714748.
- [29] J. Guo och L. Wang, "Application of Style Transfer Algorithm in Interactive Art Design of Mobile Phone Interface," *Mobile Information Systems*, 2022. doi: 10.1155/2022/7469090.
- [30] D. Ripalda, C. Guevara och A. Garrido, *Relationship Between Gestalt and Usability Heuristics in Mobile Device Interfaces* (Advances in Intelligent Systems and Computing). 2021, vol. 1269 AISC. DOI: 10.1007/978-3-030-58282-1_25.
- [31] J. N. J. Walonoski. M. Kramer, "Synthesia: An approach, method, and software mechanism for generating synthetic patients and the Synthetic Electronic Health Care Record," *Journal of the American Medical Informatics Association*, vol. 25, nr 3, s. 230–238, 2017. doi: 10.1093/jamia/ocx079.
- [32] H. L. S. International, *HL7 FHIR: Storage Module*. [Online]. Tillgänglig: <http://hl7.org/fhir/storage.html> (hämtad 2023-04-20).
- [33] MongoDB, *Horizontal vs. vertical scaling comparison guide*, [Online]. Tillgänglig: <https://www.mongodb.com/basics/horizontal-vs-vertical-scaling> (hämtad 2023-04-20).

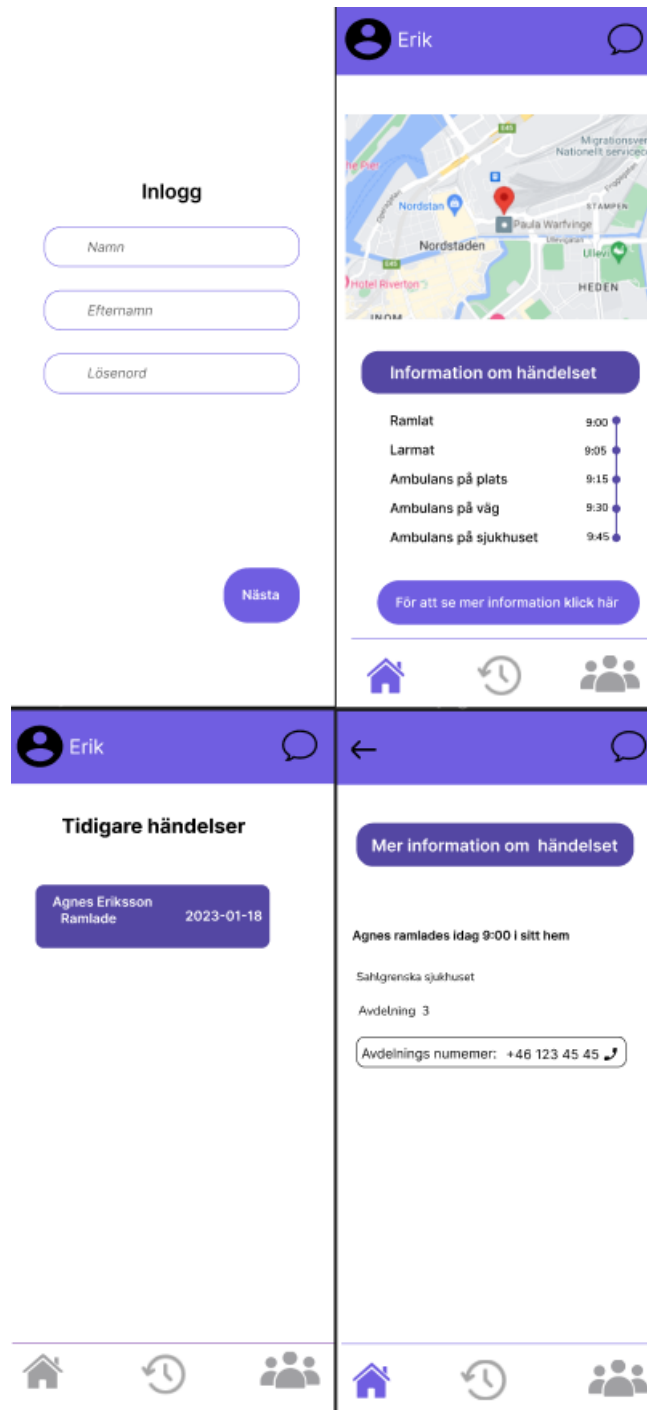
Bilagor



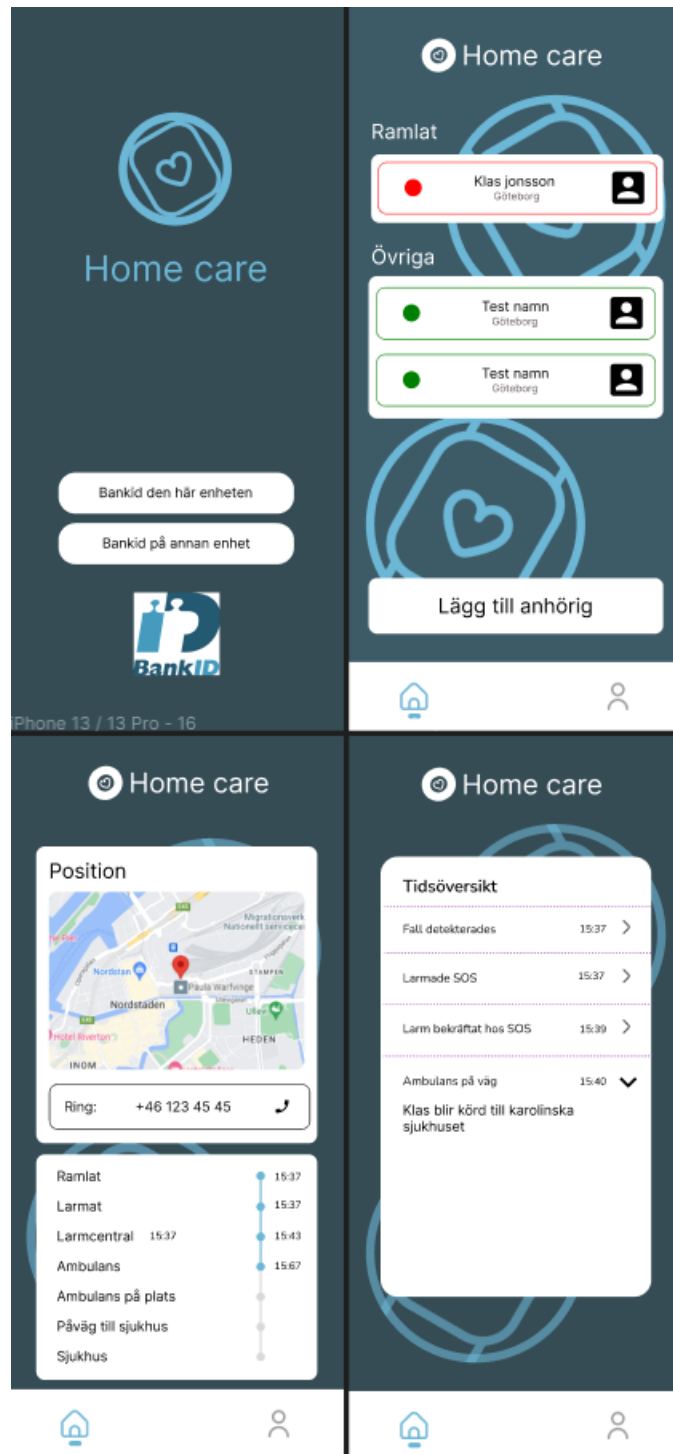
Figur 6.1: Första designen.



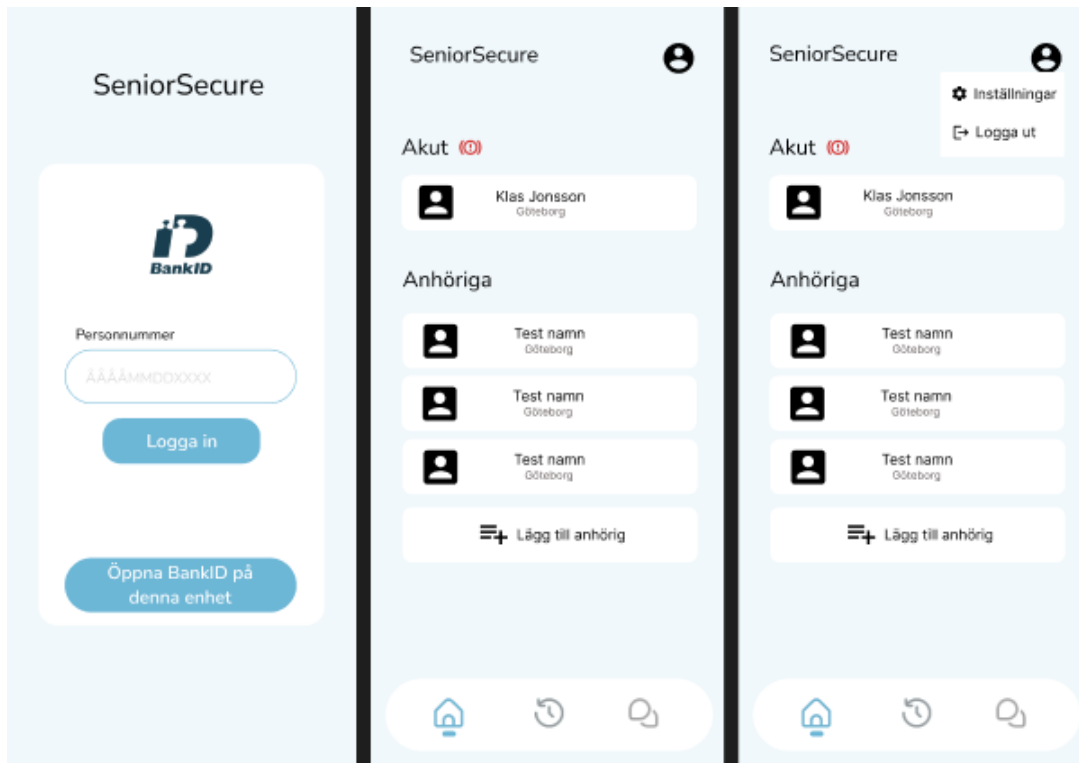
Figur 6.2: Designförslag 1.



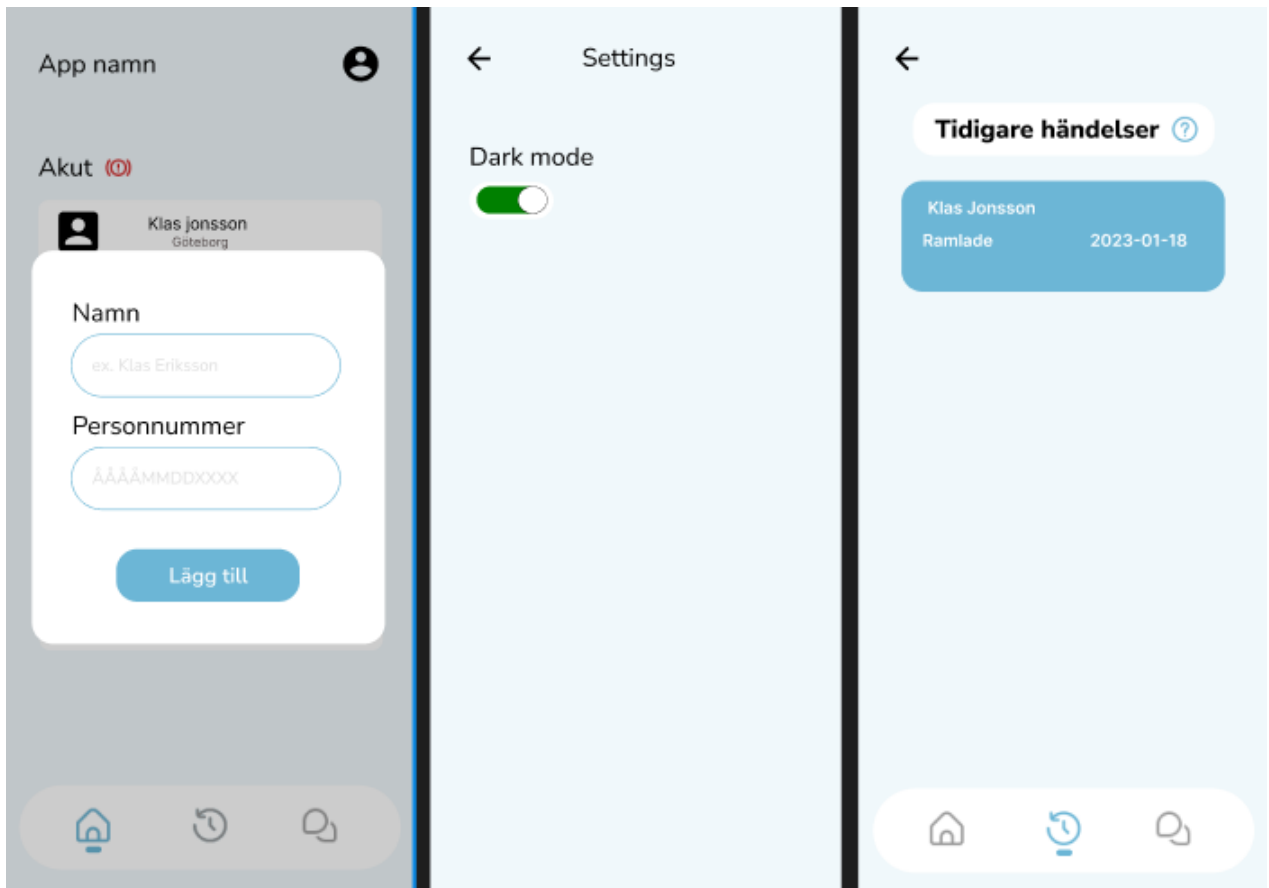
Figur 6.3: Designförslag 2.



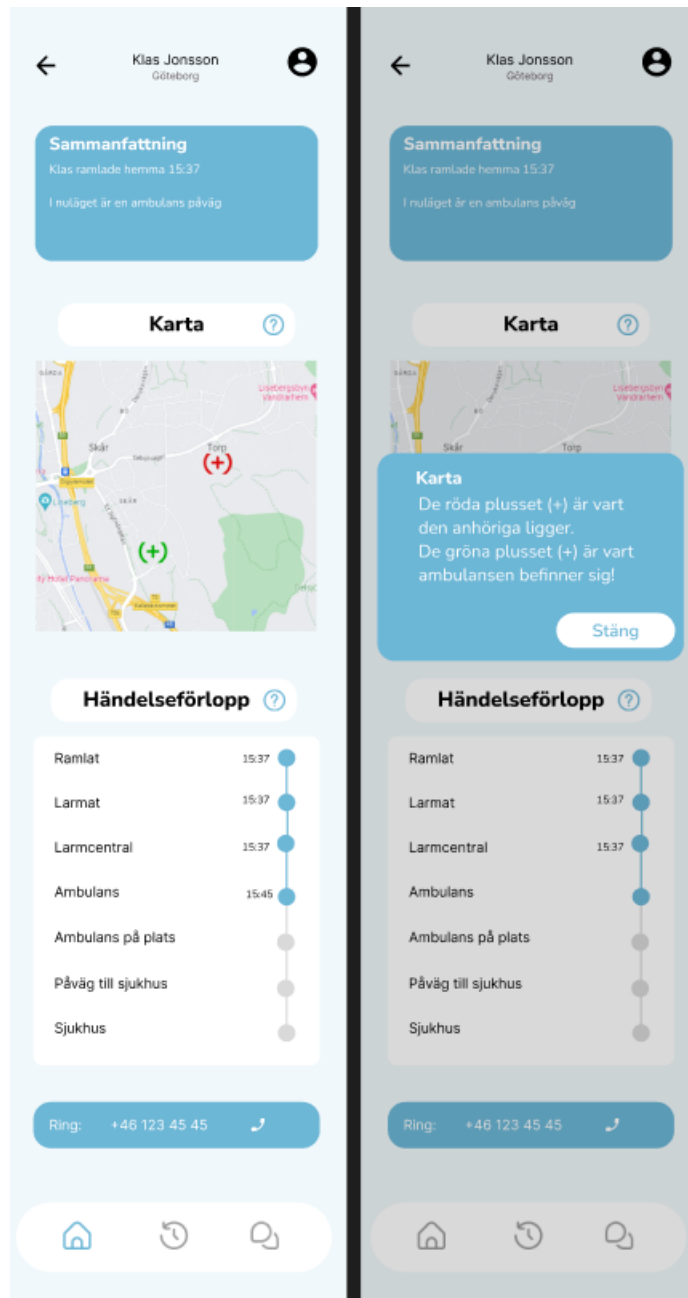
Figur 6.4: Designförslag 3.



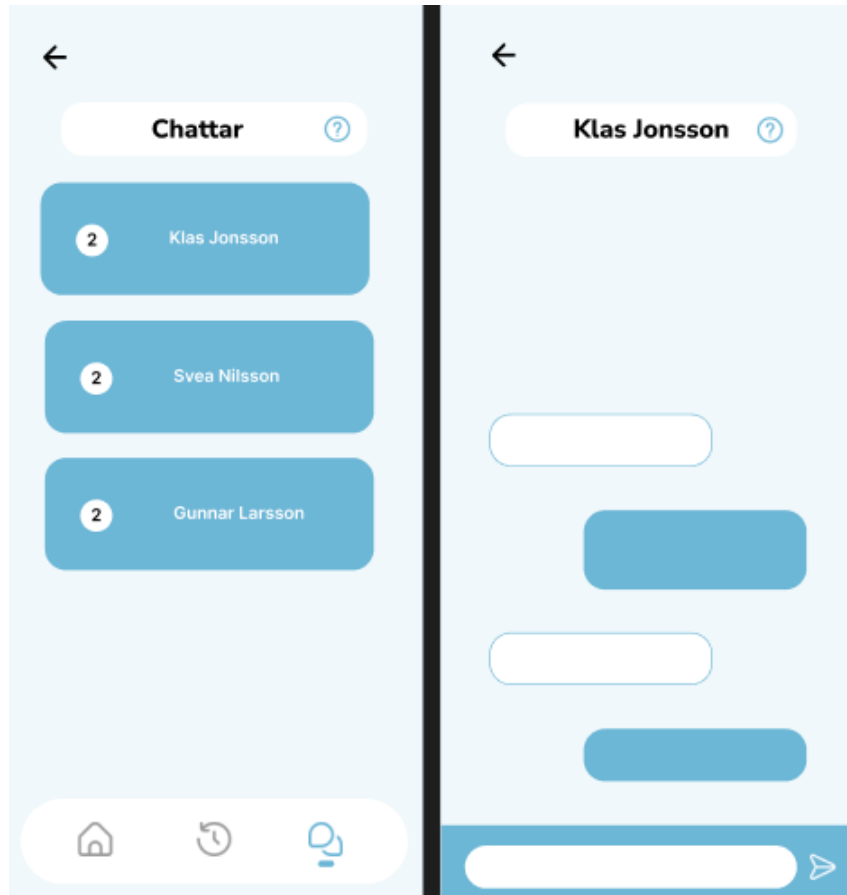
Figur 6.5: Till vänster: inloggningssidan, i mitten: startsidan och till höger demonstreras rullgardinsmenyn (jmf. eng. *dropdown menu*) uppe till höger.



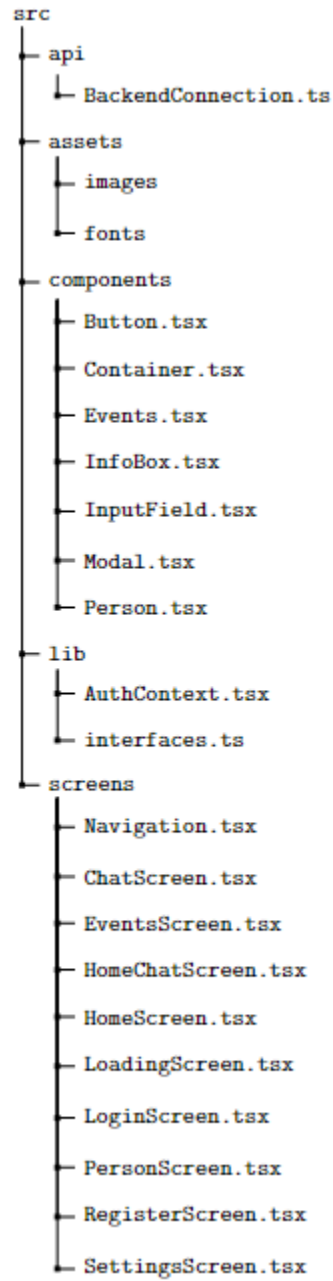
Figur 6.6: Till vänster: Startsidan efter man tryckt på lägg till ny person knappen. I mitten: inställningar-sidan och till höger: tidigare händelser-sidan.



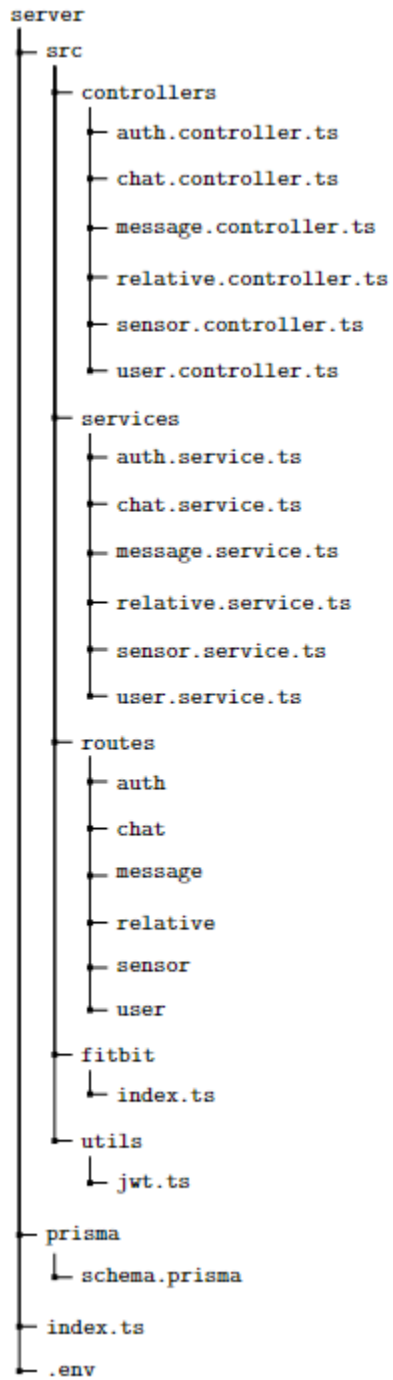
Figur 6.7: Design för att inspektera en i sekundära målgruppen som ramlat.



Figur 6.8: Design över de olika chatterna man är med i samt en layout på en chatt.



Figur 6.9: Frontend filstruktur.



Figur 6.10: Backend filstruktur.